

ADA020470

(D)

TEXAS A&M UNIVERSITY

COLLEGE STATION, TEXAS 77843

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION



DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

THEMIS

4/16/97

TEXAS A&M UNIVERSITY
PROJECT THEMIS

Technical Report No. 51

STATISTICAL PERT: AN IMPROVED SUBNETWORK
ANALYSIS PROCEDURE

by

R. L. Sielken Jr., H. O. Hartley, R. K. Spoerl

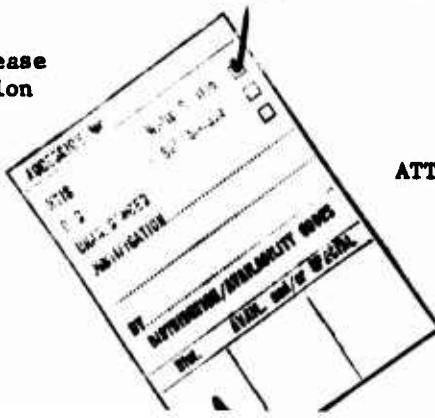
Texas A&M Research Foundation
Office of Naval Research
Contract N00014-68-A-0140
Project NRO47-700

Reproduction in whole or in part
is permitted for any purpose of
the United States Government.

Copy available to DDC does not
permit fully legible reproduction

This document has been
approved for public release
and sale; its distribution
is unlimited.

ATTACHMENT I



**STATISTICAL PERT: AN IMPROVED SUBNETWORK
ANALYSIS PROCEDURE**

by

R. L. Sielken Jr., H. O. Hartley, R. K. Spoeri

**THEMIS OPTIMIZATION RESEARCH PROGRAM .
Technical Report No. 51**

**INSTITUTE OF STATISTICS
Texas A&M University
November 1975**

Research conducted through the
Texas A&M Research Foundation
and sponsored by the
Office of Naval Research
Contract N00014-68-A-0140
Project NRO47-700

Reproduction in whole or in part
is permitted for any purpose of
the United States Government.

This document has been
approved for public release
and sale; its distribution
is unlimited.

ATTACHMENT II

TABLE OF CONTENTS

Section	Page
0 Introduction	1
1 Analysis of a Subnetwork	2
1.1 Formation of Clusters	3
1.2 Approximate Subnetwork Completion Time Moments and Distribution	7
1.2.1 A Lower Bound on T_x and an Upper Bound on F	7
1.2.2 An Upper Bound on T_r and a Lower Bound on F	12
1.2.3 Summary	13
2 Comparison with Original Subnetwork Analysis Procedure .	14
3 Conclusion	24
References	25
Appendix A: Proof of Theorem 1	26
Appendix B: New Subnetwork Analysis Program	31
Appendix C: Original Subnetwork Analysis Program . . .	65
Appendix D: Monte Carlo PERT Simulation Program	99

ABSTRACT

R. L. Sielken Jr., H. O. Hartley, R. K. Spoerl

Statistical PERT is a new procedure for obtaining information about the distribution of a project's completion time when the project is comprised of a large number of activities and the time required to complete an individual activity once it can be begun is a random variable. The project is represented as an acyclic network whose arcs correspond to the project activities. This network is simplified by replacing various activity configurations by single equivalent activities and then decomposed into several subnetworks. The distribution and moments of each subnetwork's completion time are bounded and approximated on the basis of two points from each activity's completion time distribution by using some mathematical programming techniques and a new result in the theory of networks. The project's completion time distribution is then approximated by combining the approximate subnetwork distributions.

This report describes several refinements in the subnetwork analysis procedure. One major refinement greatly reduces the computational effort in obtaining bounds on the project completion time moments and distribution. A second major refinement allows the two-point approximation of an activity's completion time distribution to better represent skewed distributions. The computer programs required to implement the new subnetwork analysis procedure are listed and documented.

Statistical PERT: An Improved Subnetwork Analysis Procedure

R. L. Sielken Jr., H. O. Hartley, R. K. Spoerl

The well-known Program Evaluation and Review Technique (PERT) is concerned with a 'project' comprised of a large number of 'activities' which are arranged as the arcs in a complex acyclic network (see e.g. Figure 1). The activities at any network node 'commence' as soon as all activities 'terminating' at that node are completed. The time required to complete an activity once it can be begun is a random variable, and hence the time needed to complete the entire project is also a random variable.

In Technical Report No. 48 "Statistical Critical Path Analysis in Acyclic Networks: Statistical PERT" a comprehensive new procedure for obtaining information on the project completion time and its distribution was described and illustrated. That procedure involved the following five general steps:

Step 1: Identification

Represent the project and its component activities in terms of an acyclic network with one source and one sink. Identify each activity's completion time distribution or at least two points on each activity's completion time distribution.

Step 2: Simplification

Replace various activity configurations and their associated completion time distributions by a single equivalent activity and completion time distribution.

Step 3: Decomposition

Decompose the simplified network into several subnetworks by separating parallel subnetworks and then separating the resulting subnetworks at each cut vertex. A cut vertex is any node such that every path from the source to the sink passes through it.

Step 4: Analysis

Each subnetwork arising from Step 3 is analyzed on the basis of two points from each component activity's completion time distribution. The result of this analysis is an approximation of each subnetwork's completion time distribution and the moments of this distribution.

Step 5: Synthesis

Combine the approximate subnetwork completion time distributions determined in Step 4. The result is an approximate completion time distribution for the entire project.

The purpose of this report is to document several refinements in the subnetwork analysis step, Step 4.

1. Analysis of a Subnetwork

The analytical procedure described in this section yields the following information on each subnetwork when each component activity's completion time distribution is replaced by a discrete two-point distribution:

- (a) Upper and lower bounds on the mean subnetwork completion time as well as the other moments of the subnetwork completion time.
- (b) Upper and lower bounds on the distribution function of the subnetwork completion time.

- (c) An approximate distribution function of the subnetwork completion time.

Each subnetwork is assumed to be an acyclic network with one source, one sink, and no cut vertices.

The analysis of each subnetwork involves essentially two parts:

1. The formation of "clusters" of activities whose effect on the subnetwork completion time seems to be interrelated.
2. The approximation of the subnetwork completion time moments and distribution on the basis of the clusters.

1.1 Formation of Clusters

The actual completion time distribution of each individual activity, A, in the subnetwork is replaced by a discrete distribution with probability P at the lower point, ℓ_A , and $Q = 1 - P$ at the upper point, u_A .

Let n be the number of activities in the subnetwork. Then for each of the 2^n combinations of the ℓ_A 's and u_A 's there will be a subnetwork completion time (a critical path time). The r-th moment of these 2^n times will be denoted by T_r , and the distribution function of these times will be denoted by F. The approximation of the T_r 's (especially T_1 , the mean) and F is the goal of the subnetwork analysis. Since n will usually be fairly large, the complete enumeration of the 2^n critical path times will usually be unreasonable. Hence the activities which are most likely to be on the critical path through the subnetwork are identified and their joint behavior investigated.

The mean of the completion time distribution for activity A is defined to be $m_A = P\ell_A + Qu_A$. The standard deviation of the completion time distribution for activity A is defined to be $s_A = \sqrt{P\ell_A^2 + Qu_A^2 - m_A^2}$

and is assumed to be positive. (The assumption that $s_A > 0$ is not really a practical restriction since the difference between a fixed activity completion time and one with a very small dispersion is negligible from a practical viewpoint.) The subnetwork's critical path when each activity's completion time is set equal to its mean will be referred to as the "original" critical path. The activities on this critical path will be referred to as "critical activities" with K equalling the number of such activities. Some non-critical activities might become critical if some of the completion times for the original critical activities were decreased. These activities are identified as follows. The completion time for one critical activity, say A , is set equal to $\max\{m_A - \lambda s_A, 0\}$ where λ is a non-negative algorithm parameter which the user specifies. All other completion times are set equal to their means. Then the longest path through the resulting network is determined. Any activities on this path which were not on the original critical path are now referred to as the "associates" of A since the effect of these "associates" on the networks' completion time is related to A 's completion time. This procedure is repeated for each original critical activity.

Each critical activity and its associates make up one "cluster". These K initial clusters are now "pooled" by combining any two clusters with at least one activity in common. In general there will still be more than one cluster, and many of the $n - K$ non-critical activities will not occur in any cluster.

The associates correspond to the activities which become critical when the completion times of the original critical activities are lowered. However, some of the originally non-critical activities may also become critical if their completion times exceed their means

and the completion times of the original critical activities are at their means. These activities are identified next. Each originally non-critical activity is investigated separately. If activity A is being investigated, then the completion time for A is set equal to $m_A + \theta s_A$ where θ is a non-negative algorithm parameter which the user specifies. The completion times for all other activities are set equal to their means, and the corresponding critical path determined. This critical path will either be the original critical path or a new path which includes A. In the latter case, the activities on the original critical path which are not on a new critical path containing A are called the "eliminants" of A. Thus, the effect of A's eliminants on the networks completion time is related to the completion time for A. Hence, A is added to any cluster which contains at least one of A's eliminants. After this procedure has been repeated for each originally non-critical activity, the resultant clusters are "pooled" again by combining any two clusters with at least one activity in common.

Although the number of clusters is reduced when the pooling on the basis of the associates occurs and then further reduced when the pooling on the basis of the eliminants occurs, there will generally remain more than one cluster and several activities not in any cluster.

In general the larger the values of λ and θ the greater the number of activities in the clusters and the smaller the number of clusters. In particular the procedure for forming the clusters has the following properties:

Property 1: If $\lambda_2 > \lambda_1$, then any activity which would be an associate of a critical activity A when $\lambda = \lambda_1$ would also be an associate of A when $\lambda = \lambda_2$.

Property 2: If $\theta_2 > \theta_1$, then any critical activity which would be an eliminant of a non-critical activity A when $\theta = \theta_1$ would also be an eliminant of A when $\theta = \theta_2$.

Property 3: For any originally non-critical activity A there exists θ_A such that A will have some eliminants for any $\theta \geq \theta_A$.

Property 4: For any fixed value of λ , the set of activities in the union of the clusters is monotonically non-decreasing as $\theta \rightarrow \infty$.

Property 5: There exists a finite value θ^* such that if $\theta \geq \theta^*$, then every activity would be in some cluster.

Property 6: The number of clusters, originally K, is non-increasing as $\theta \rightarrow \infty$.

Property 7: There exists a finite value θ^* such that if $\theta \geq \theta^*$, then there would only be one cluster.

Most of the properties of the cluster formation procedure are fairly straightforward; however, Property 7 requires some special justification. This justification is based on the following definition and theorem which is proven in Appendix A.

Definition: In any acyclic network a bridge over any two consecutive arcs A_1 and A_2 is any arc A_3 such that all paths from the source to the sink passing through A_3 do not pass through either A_1 or A_2 .

Theorem 1: In any acyclic network with no cut vertices there is at least one bridge for any pair of consecutive arcs.

Property 3 implies that all activities will belong to some cluster if $\theta \geq \theta^*$ and

$$\theta^* = \max_A \{\theta_A : A \text{ originally non-critical}\}.$$

Now consider any two consecutive activities A_1 and A_2 on the original critical path. Theorem 1 implies that there is a bridge over A_1 and A_2 , say A_3 . Since the original critical path passes through A_1 and A_2 , A_3 cannot be on the original critical path. Therefore, if $\theta \geq \theta^* \geq \theta_{A_3}$, A_1 and A_2 will be eliminants of A_3 and hence will be in the same cluster as A_3 . Thus, since each cluster contains at least one original critical activity and any two consecutive original critical path activities belong to the same cluster when $\theta \geq \theta^*$, there is only one cluster when $\theta \geq \theta^*$ and Property 7 is established.

1.2 Approximate Subnetwork Completion Time Moments and Distribution

1.2.1 A Lower Bound on T_r and an Upper Bound on F

For each cluster C let n_c denote the number of activities in C, and let $v = 1, \dots, 2^{n_c}$ index the 2^{n_c} configurations of activity completion times when

- (a) the completion time for each activity A not in C is equal to its lower point, ℓ_A , and
- (b) the completion times for the activities in C are at each of the 2^{n_c} possible combinations of their upper and lower points.

Let

t_v = critical path time for the v-th configuration

and

p_v = probability of the v-th configuration

$$= \prod_{i=1}^{n_c} [P_i(1 - \delta_{v,i}) + Q_i \delta_{v,i}]$$

where

$\delta_{v,i} = 1$ if the time for the i -th activity in C is u_i in
the v -th configuration
 $= 0$ if the time for the i -th activity in C is t_i in the
 v -th configuration.

Then

$$\hat{T}_r^-(C) \equiv \sum_{v=1}^{n_c} p_v t_v^r$$

and

$$\hat{F}^+(t; C) \equiv \sum_{v=1}^{n_c} p_v I_t(t_v)$$

where

$$I_t(t_v) = 1 \quad \text{if } t_v \leq t \\ = 0 \quad \text{if } t_v > t$$

are the r -th moment of the n_c critical path times and their distribution function respectively. Let

$$T_r^-(\theta, \lambda) = \max_C \hat{T}_r^-(C)$$

and

$$F^+(t; \theta, \lambda) = \min_C \hat{F}^+(t; C)$$

which depend on θ and λ since the composition and number of clusters depend on θ and λ .

The first step in showing that $T_r^-(\theta, \lambda)$ is a lower bound for T_r is proving the following theorem:

Theorem 2: For any cluster C , any positive integer r , and any activity A not in C ,

$$\hat{T}_r^-(C \cup \{A\}) \geq \hat{T}_r^-(C).$$

Proof: Consider any particular critical path for a particular one of the 2^{n_c} combinations of upper and lower points involved in $\hat{T}_r^-(C)$. Consider the following two cases:

- (i) activity A with its completion time equal to l_A is on the critical path, and
- (ii) activity A with its completion time equal to l_A is not on the critical path.

Let $\delta_A = (u_A - l_A)$ and the particular critical path time be t . Then in case (i)

- (a) if the completion time for A is set equal to u_A , this will increase the r-th moment of the critical path time by $Q[(t + \delta_A)^r - t^r]$; and
- (b) if the completion time for A is set equal to l_A , this will not alter the r-th moment of the critical path time.

In case (ii)

- (a) if the completion time for A is set equal to u_A , this may increase the r-th moment of the critical path time by $Q[(t + \delta_A)^r - t^r]$ or less, and
- (b) if the completion time for A is set equal to l_A , this will not alter the r-th moment of the critical path time.

Therefore in either case the contribution to $\hat{T}_r^-(C \cup \{A\}) - \hat{T}_r^-(C)$ for this particular critical path will be between 0 and $Q[(t + \delta_A)^r - t^r]$. Since the contribution is non-negative for each particular combination,

$$\hat{T}_r^-(C \cup \{A\}) \geq \hat{T}_r^-(C).$$

QED

A straightforward application of Theorem 2 yields the following theorem:

Theorem 3: For any two clusters C_1 and C_2 and any positive integer r ,

$$\hat{T}_r^-(C_1 \cup C_2) \geq \max\{\hat{T}_r^-(C_1), \hat{T}_r^-(C_2)\}.$$

Property 2 of the cluster formation procedure implies that if θ is increased the clusters expand or are pooled. Thus, Theorems 2 and 3 imply that, for fixed λ , $T_r^-(\theta, \lambda)$ is non-decreasing as θ increases. Furthermore, Properties 5 and 7 together imply that for θ sufficiently large there is only one cluster and all of subnetwork activities are in that cluster. Hence, for θ sufficiently large $T_r^-(\theta, \lambda) = T_r^-$, and the following theorem is true:

Theorem 4:

- (a) $T_r^-(\theta, \lambda)$ is a non-decreasing function of θ for any fixed values of λ and r ;
- (b) there exists a finite value θ^* such that $\theta \geq \theta^*$ implies

$$T_r^-(\theta, \lambda) = T_r^-$$

for any λ , and r ; and

- (c) for any θ , λ , and r

$$T_r^-(\theta, \lambda) \leq T_r^+.$$

Similarly, the first step in showing that $F^+(t; \theta, \lambda)$ is an upper bound on $F(t)$ is the following theorem:

Theorem 5: For any cluster C , any value of t , and any activity A not in C ,

$$\hat{F}^+(t; C \cup \{A\}) \leq \hat{F}^+(t; C).$$

Proof: Consider any particular configuration of activity times used to determine $F^+(t; C)$ before C is augmented by A . When A is added to C , this particular configuration will appear once with A at its upper percentile and once with A at its lower percentile. When A is at its lower percentile, the configuration's critical path time is unchanged. However, when A is at its upper percentile, the configurations' critical path time is either unchanged or possibly increased if A were on the configuration's critical path. Thus, the addition of A leaves the cumulative probability associated with critical path times less than or equal to t either unchanged or decreased. QED

A straightforward extension of Theorem 5 is the following theorem:

Theorem 6: For any two clusters C_1 and C_2 and any t ,

$$\hat{F}^+(t; C_1 \cup C_2) \leq \min\{\hat{F}^+(t; C_1), \hat{F}^+(t; C_2)\}.$$

Since Property 2 of the cluster formation procedure implies that the clusters expand or are pooled if θ is increased, Theorem 5 and Theorem 6 together imply that, for all t and any fixed λ , $F^+(t; \theta, \lambda)$ is non-increasing function of θ . Furthermore, Properties 5 and 7 together imply that for θ sufficiently large there is only one cluster and all of the subnetwork activities are in that cluster. Hence, for θ sufficiently large $F^+(t; \theta, \lambda) = F(t)$ and the following theorem is true:

Theorem 7:

- (a) $F^+(t; \theta, \lambda)$ is a non-increasing function of θ for every t and any λ ;

- (b) there exists a finite value θ^* such that $\theta \geq \theta^*$ implies

$$F^+(t; \theta, \lambda) = F(t)$$

for every t and λ ; and

- (c) for any θ , λ , and t

$$F^+(t; \theta, \lambda) \geq F(t).$$

1.2.2 An Upper Bound on T_r^+ and a Lower Bound on F

For each cluster C let n_C denote the number of activities in C , and let $v = 1, \dots, 2^{n_C}$ index the 2^{n_C} configurations of activity completion times when

- (a) the completion time for each activity not in the cluster is equal to its upper point, and
- (b) the completion times for the activities in the cluster are at each of the 2^{n_C} possible combinations of their upper and lower percentiles.

Let t_v , p_v , and $I_t(t_v)$ be as before and define

$$\hat{T}_r^+(C) = \sum_{v=1}^{2^{n_C}} p_v t_v^r,$$

$$T_r^+(\theta, \lambda) = \min_C \hat{T}_r^+(C)$$

$$\hat{F}^-(t; C) = \sum_{v=1}^{2^{n_C}} p_v I_t(t_v),$$

and

$$F^-(t; \theta, \lambda) = \max_C \hat{F}^-(t; C).$$

Then an argument completely analogous to that used to prove Theorem 4 and Theorem 7 leads to the following theorems:

Theorem 8:

- (a) $T_r^+(\theta, \lambda)$ is a non-increasing function of θ for any fixed values of λ and r ;
- (b) there exists a finite value θ^* such that $\theta \geq \theta^*$ implies

$$T_r^+(\theta, \lambda) = T_r$$

for any λ and r ; and

- (c) for any θ , λ , and r

$$T_r \leq T_r^+(\theta, \lambda).$$

Theorem 9:

- (a) There exists a finite value θ^* such that $\theta \geq \theta^*$ implies

$$F^-(t; \theta, \lambda) = F(t)$$

for every t and any λ ; and

- (b) for any θ , λ , and t

$$F^-(t; \theta, \lambda) \leq F(t).$$

1.2.3 Summary

Theorems 2-9 together imply that for any value of θ and λ chosen by the algorithm user:

- (a) $F^-(\theta, \lambda) \leq T_r \leq T_r^+(\theta, \lambda)$, for any positive integer r ; and
- (b) $F^-(t; \theta, \lambda) \leq F(t) \leq F^+(t; \theta, \lambda)$ for any t .

They also imply that, for any r , λ , and t ,

$$T_r^+(\theta, \lambda) = T_r^-(\theta, \lambda)$$

and

$$F^+(t; \theta, \lambda) = F^-(t; \theta, \lambda)$$

would decrease monotonically to zero if θ were increased. In fact, Theorems 2-9 imply that there exists a value θ^* which doesn't depend on r , λ , or t such that $\theta \geq \theta^*$ implies that

$$T_r^-(\theta, \lambda) = T_r = T_r^+(\theta, \lambda)$$

and

$$F^-(t; \theta, \lambda) = F(t) = F^+(t; \theta, \lambda).$$

A reasonable approximation for $F(t)$ is

$$\hat{F}(t; \theta, \lambda) = \frac{1}{2}[F^-(t; \theta, \lambda) + F^+(t; \theta, \lambda)].$$

2. Comparison with the Original Subnetwork Analysis Procedure

The original subnetwork analysis procedure documented in Technical Report No. 48 formed the clusters in essentially the same way as the new subnetwork analysis procedure described in this report except that in the original procedure P always equalled $\frac{1}{2}$ and θ and λ multiplied the point difference, $u_A - l_A$, instead of the standard deviation, $\sqrt{Pl_A^2 + Qu_A^2 - m_A^2}$.

In the original procedure $n_U = \sum_c n_c$ denotes the number of activities in the union of the clusters, and $T_r^+(\theta, \lambda)$ is defined to be the average of the r -th power of the 2^{n_U} critical path times when

- (a) the completion time for each activity not in the union of the clusters is equal to its upper point, and

- (b) the completion times for the activities in the union of the clusters are at each of the 2^{n_U} possible combinations of their upper and lower points.

Correspondingly, $F^-(t; \theta, \lambda)$ was defined to be the proportion of these 2^{n_U} critical path times that were less than or equal to t . Analogously, $F^+(t; \theta, \lambda)$ was the proportion of the 2^{n_U} critical path times less than or equal to t when

- (a) the completion time for each activity not in the union of the clusters is equal to its lower point, and
- (b) the completion times for the activities in the union of the clusters are at each of the 2^{n_U} possible combinations of their upper and lower points.

The only problem with this procedure is that 2^{n_U} may be quite large even for relatively small values of (θ, λ) . For example, if the original critical path contains 10 activities and each critical activity has one associate, then $2^{n_U} = 2^{20} = 1,048,576$, and the determination of $T_r^+(\theta, \lambda)$, $F^-(t; \theta, \lambda)$, and $F^+(t; \theta, \lambda)$ requires the evaluation of over 2 million critical path times. On the other hand, in the new procedure the determination of $T_r^-(\theta, \lambda)$, $T_r^+(\theta, \lambda)$, $F^-(t; \theta, \lambda)$, $F^+(t; \theta, \lambda)$ requires the evaluation of only $2 \sum_c 2^{n_c}$ critical path times, 80 in the example. Thus the new procedure greatly reduces the computational effort required to bound the project completion time moments and distribution.

A practical alternative to evaluating all 2^{n_U} critical path times called for in the original procedure is to randomly sample the 2^{n_U} critical path times when n_U is large and base the bounds on the sample critical path times. A computer program for the original procedure with a sampling option is documented in Appendix C. (This program supercedes

the subnetwork analysis program given in Technical Report No. 48.) It should be noted that the original subnetwork analysis procedure with the sampling option in effect is still superior to a simple Monte Carlo simulation of the subnetwork since the subnetwork analysis procedure

- (i) provides the information in terms of associates and eliminants about which activities play an important role in determining the subnetwork's completion time and the interactions among activities, and
- (ii) samples only those activities which most influence the subnetwork completion time,

Furthermore, the loss in accuracy due to sampling in the subnetwork analysis procedure seems to be quite minimal even for relatively small sample sizes - see for example Table 2 and Table 3.

All specific computational results documented in this report are for the project network given in Figure 1. (This network is the simplified network from a large naval PERT problem.) A listing of each activity's two-point approximation except for its P and Q is given in Table 1.

Sampling can also be used in the new subnetwork analysis procedure in the somewhat unlikely event that θ and λ are chosen so large that 2^{n_c} for some cluster C is too large. In this case the probability that the v-th configuration of activity completion times $v = 1, \dots, 2^{n_c}$ is selected on a sampling trial is

$$p_v = \prod_{i=1}^{n_c} [P_i(1 - \delta_{v,i}) + Q_i \delta_{v,i}]$$

where

Figure 1. A Project Network

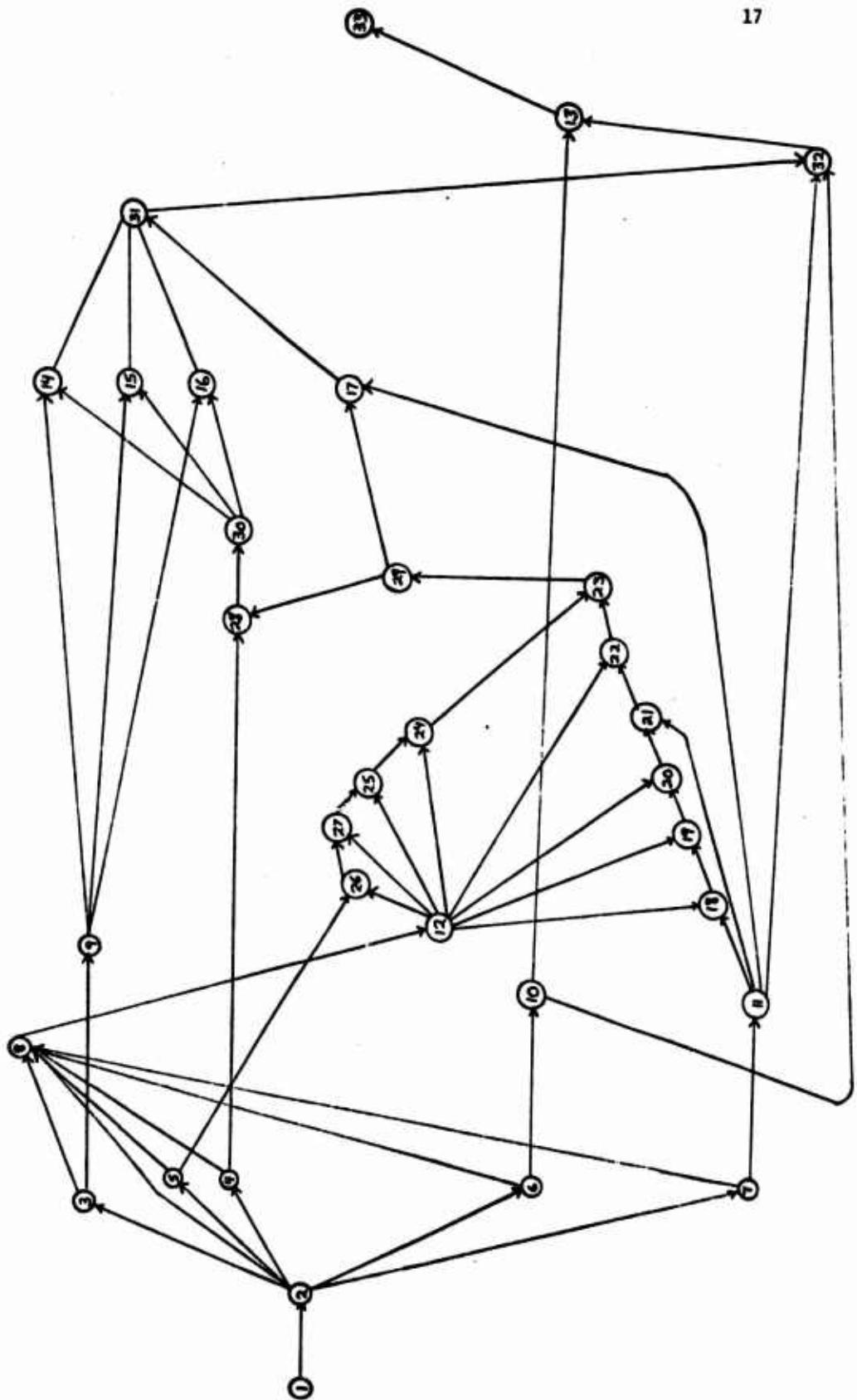


Table 1. The Upper and Lower Points in the Two-Point Approximations to the Activity Completion Time Distributions for the Project Network in Figure 1

Activity Number	Origin Node	Terminal Node	Lower Point	Upper Point
1	1	2	0.0	0.0
2	8	12	336.27	429.47
3	2	3	57.47	89.96
4	2	4	57.47	89.96
5	2	5	57.47	89.96
6	2	6	57.47	89.96
7	2	7	57.47	89.96
8	3	8	68.96	107.95
9	4	8	68.96	107.95
10	5	8	68.96	107.95
11	6	8	68.96	107.95
12	7	8	68.96	107.95
13	2	8	150.36	193.49
14	6	10	333.96	403.85
15	3	9	333.96	403.85
16	7	11	355.10	409.85
17	11	18	141.75	221.90
18	10	13	672.36	783.11
19	9	14	560.89	660.00
20	9	15	560.89	660.00
21	9	16	560.89	660.00
22	11	17	542.80	638.71
23	12	18	111.10	173.92
24	18	19	256.03	346.98
25	12	19	302.80	400.67
26	12	20	311.95	410.71
27	11	21	423.58	530.74
28	12	22	315.35	415.71
29	19	20	7.66	11.99
30	20	21	16.77	22.55
31	21	22	11.49	17.99
32	22	23	39.54	48.87
33	12	26	301.91	400.86
34	5	26	767.09	892.74
35	12	27	350.31	460.06
36	12	24	382.32	464.98
37	12	25	385.28	461.54
38	25	24	11.49	17.99
39	24	23	16.28	23.00
40	26	27	7.66	11.99
41	27	25	20.86	28.29
42	4	28	810.17	976.10
43	23	29	15.32	23.99

Activity Number	Origin Node	Terminal Node	Lower Point	Upper Point
44	28	30	15.32	23.99
45	14	31	57.47	89.96
46	16	31	49.81	77.96
47	15	31	53.64	83.96
48	17	31	88.12	137.94
49	31	32	3.83	6.00
50	32	13	49.81	77.96
51	13	33	109.01	152.63
52	11	32	745.50	811.32
53	10	32	714.74	799.83
54	30	14	0.0	0.0
55	30	16	0.0	0.0
56	30	15	0.0	0.0
57	29	17	0.0	0.0
58	29	28	0.0	0.0

Table 2. The Effect of Sampling in the Original Subnetwork Analysis Procedure: Percentiles of the Project Completion Time Distribution

Percentiles	Sample Sizes							
	$(\theta, \lambda) = (0, 0)$			$(\theta, \lambda) = (.25, 0)$				
	$n_U = 2^6$	$n_U = 2^5$	$n_U = 2^4$	$n_U = 2^{11}$	1000	500	200	100
.05	1360	1363	1377	1381	1381	1381	1384	1384
.10	1383	1390	1413	1409	1406	1411	1406	1414
.15	1403	1413	1423	1424	1421	1429	1424	1423
.20	1416	1423	1436	1441	1436	1443	1441	1446
.25	1430	1430	1446	1451	1448	1451	1451	1453
.30	1446	1453	1466	1458	1456	1461	1458	1463
.35	1459	1479	1472	1468	1466	1471	1468	1471
.40	1476	1486	1476	1478	1473	1478	1476	1478
.45	1499	1499	1512	1483	1483	1483	1483	1483
.50	1542	1542	1516	1488	1486	1491	1491	1488
.55	1578	1578	1575	1496	1496	1498	1498	1498
.60	1578	1578	1575	1501	1501	1505	1501	1503
.65	1602	1602	1579	1513	1510	1513	1513	1513
.70	1605	1605	1602	1515	1515	1515	1523	1515
.75	1605	1618	1605	1525	1525	1525	1528	1525
.80	1621	1618	1605	1533	1533	1530	1538	1530
.85	1622	1622	1618	1543	1543	1543	1545	1543
.90	1645	1622	1622	1558	1558	1558	1570	1577
.95	1648	1645	1648	1585	1585	1582	1587	1597
.975	1648	1648	1648	1607	1607	1605	1605	1622
.99	1648	1648	1648	1625	1625	1625	1622	1647
1.00	1648	1648	1648	1649	1649	1649	1649	1649

Table 3. The Effect of Skewed Activity Completion Time Distributions on the Subnetwork Analysis Procedures.

Estimated Percentiles of the Project Completion Time Distribution

Percentile	Monte Carlo	Original Subnetwork Analysis Procedure (Sample Size = 1000/cluster)		New Subnetwork Analysis Procedure (Sample Size = 1000/cluster)	
		1st Run	2nd Run	1st Run	2nd Run
.05	1405	1402	1396	1370	1374
.10	1423	1416	1416	1390	1398
.15	1435	1430	1433	1402	1410
.20	1447	1438	1443	1425	1437
.25	1457	1448	1449	1437	1449
.30	1467	1453	1456	1450	1453
.35	1477	1462	1466	1461	1461
.40	1486	1469	1473	1484	1500
.45	1496	1476	1479	1504	1524
.50	1507	1481	1483	1528	1528
.55	1516	1487	1489	1528	1528
.60	1527	1495	1496	1528	1528
.65	1538	1500	1499	1563	1560
.70	1549	1510	1512	1567	1567
.75	1560	1517	1516	1579	1579
.80	1580	1526	1529	1587	1587
.85	1603	1541	1542	1598	1602
.90	1627	1550	1552	1610	1614
.95	1675	1575	1582	1622	1673
1.00	2019	1646	1648	1720	1720

*The "1st run" and "2nd run" correspond to two different samples with different initializations of the random number generator.

$\delta_{v,i} = 1$ if the time for the i -th activity in C is u_i in
the v -th configuration
 $= 0$ if the time for the i -th activity in C is ℓ_i in
the v -th configuration.

Then with $I_t(t_v)$ as before the estimated bounds from C are

$$\hat{T}_r^-(C) = \frac{2^{n_c}}{N} \sum_{v=1}^{n_c} w_v p_v t_v^r$$

and

$$\hat{F}^+(t; C) = \frac{2^{n_c}}{N} \sum_{v=1}^{n_c} w_v p_v I_t(t_v)$$

where w_v is the number of times the v -th configuration appears in the sample, N is the sample size, and t_v is the critical path time corresponding to the v -th configuration of activity completion times with the completion time for each activity A not in C equal to ℓ_A . Similar modifications are made for $T_r^+(C)$ and $F^-(t; C)$. The corresponding estimators $T_r^-(\theta, \lambda)$, $T_r^+(\theta, \lambda)$, $F^-(t; \theta, \lambda)$, and $F^+(t; \theta, \lambda)$ are all unbiased.

A computer implementation of the new subnetwork analysis procedure including the sampling option is documented in Appendix B. In addition to reducing the computational effort, the new subnetwork analysis procedure allows the user to specify any probabilities (P, Q) for (ℓ, u) instead of requiring $(1/2, 1/2)$. This would not be of great significance if all activity completion time distributions were symmetric. However, since many completion time distributions are skewed, the ability to specify (P, Q) can be a real advantage. To exemplify this advantage, the completion time for each activity in Figure 1 was taken to be a linearly

transformed chi-square random variable, $(\chi_3^2 - c_1)/c_2$, where c_1 and c_2 were determined so that the points in Table 1 corresponded to the 15-th and 85-th percentiles respectively. This made the activity's completion time distribution highly skewed. Then the corresponding project completion time distribution was approximated using

- (i) a Monte Carlo simulation of size 1000,
- (ii) the new subnetwork analysis procedure with $(P, Q) = (1/2, 1/2)$ for all activities, and
- (iii) the new procedure with each activity's (P, Q) chosen so that the mode and first two moments of its two-point approximation equalled the mode and first two moments of its transformed chi-square distribution.

The results are given in Table 3. If the Monte Carlo approximation is used as a basis for comparison, the value of being able to specify (P, Q) is obvious.

The Monte Carlo PERT simulation program used in the above experiment is documented in Appendix D and supercedes the Monte Carlo PERT program given in Technical Report No. 48.

In the special case where every activity's (P, Q) is $(1/2, 1/2)$, the $\hat{T}_r^-(C)$ and $\hat{F}^+(t; C)$ can be computed on the basis of the $2^n c$ critical path times corresponding to

- (a) the completion time for each activity A not in C being equal to the mean, m_A , and
- (b) the completion times for the activities in C being at each of the $2^n c$ possible combinations of their upper and lower points.

This is a change from the usual computation in that the activities not in C are at their means here instead of their lower points. This change

will tend to improve the estimators $\bar{T}_r(\theta, \lambda)$ and $\bar{F}^+(t; \theta, \lambda)$. However, this change is only guaranteed not to invalidate Theorems 4 and 7 when all (P, Q) are $(1/2, 1/2)$. The computer implementation of the new subnetwork analysis procedure includes the option to make this change.

3. Conclusion

The new subnetwork analysis procedure

- (i) forms associates, eliminants, and clusters in essentially the same way as the original procedure,
- (ii) bounds the project's completion time moments and distribution primarily on the basis of the individual clusters instead of on a pooled cluster, and
- (iii) approximates an activity's completion time distribution by a two-point distribution with possibly unequal probabilities for the two points instead of always equal probabilities.

The advantage of (ii) is that much much fewer critical path times need to be evaluated in determining the bounds on the project completion time. The advantage to (iii) is the ability to better approximate skewed activity completion time distributions.

Computer implementations of the new subnetwork analysis procedure, the original subnetwork analysis procedure, and a Monte Carlo simulation algorithm are documented in Appendices B, C, and D respectively. Both subnetwork analysis procedures include options to use sampling for large clusters.

References

Sielken, R. L. Jr., L. J. Ringer, H. O. Hartley, and E. Arseven,
"Statistical Critical Path Analysis in Acyclic Stochastic
Networks: Statistical PERT," Institute of Statistics, Texas
A&M University Project Themis Technical Report No. 48,
November 1974.

APPENDIX A

Proof of Theorem 1

The principle objective of this appendix is to prove the following theorem:

Theorem 1: In any acyclic network with no cut vertices there is at least one bridge for any pair of consecutive arcs.

The networks considered in this appendix are assumed to be acyclic, have no cut vertices, and have one source and one sink. Also the two arcs A_1 and A_2 are any two adjacent (consecutive) arcs with A_1 preceding A_2 .

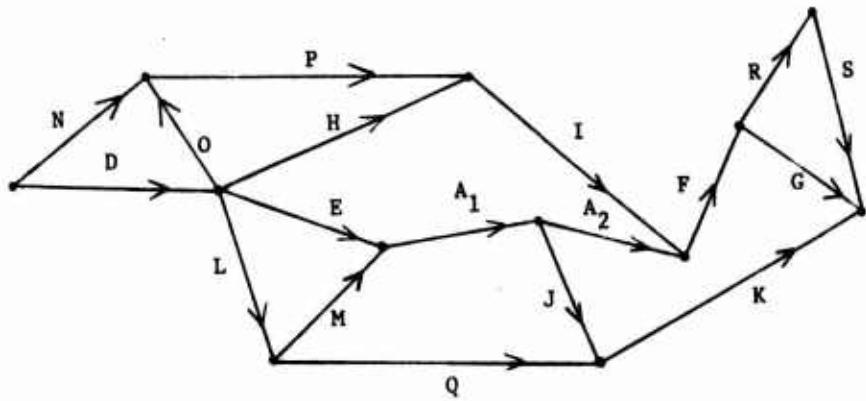
Definition 1: A bridge over A_1 and A_2 is any arc, say A_3 , such that all paths from the source to the sink passing through A_3 do not pass through either A_1 or A_2 .

Definition 2: An origin violator of A_1 and A_2 is an arc, say A_3 , such that there exists a path from the terminal node of A_1 to the sink which passes through A_3 .

Definition 3. A terminal violator of A_1 and A_2 is an arc, say A_3 , such that there exists a path from the source to the terminal node of A_1 which passes through A_3 .

An intuitive feeling for these definitions can be obtained by considering any path P^* from the source to the sink which passes through A_1 and A_2 . If an arc A_3 is an origin violator, then there is a path from the source to the sink which follows along P^* through the terminal

node of A_1 and then goes through A_3 . This path "originates" from P^* too late for A_3 to be a bridge over A_1 and A_2 . Similarly, if A_3 is a terminal violator, then there is a path from the source to the sink which passes through A_3 and then joins into P^* before P^* passes through the terminal node of A_1 . This path "terminates" into P^* too early for A_3 to be a bridge.



BRIDGES over A₁ and A₂: H, I, N, O, P, and Q

ORIGIN VIOLATORS of A₁ and A₂: A₂, F, G, J, K, R, and S

TERMINAL VIOLATORS of A₁ and A₂: A₁, D, E, L, and M

The following three lemmas are straightforward consequences of the definitions of a bridge, an origin violator, and a terminal violator.

Lemma 1: Every branch in the network is either a bridge over A_1 and A_2 , an origin violator of A_1 and A_2 , or a terminal violator of A_1 and A_2 .

Proof of Lemma 1: Suppose that A_3 is not a bridge. Then there exists a path P from the source to the sink which contains A_3 and either A_1 or A_2 .

Suppose that P contains A_1 . If $A_3 = A_1$ or A_3 precedes A_1 on P , then P contains a path from the source to the terminal node of A_1 which passes through A_3 , and A_3 would be a terminal violator. On the other hand, if A_3 follows A_1 on P , then P contains a path from the terminal node of A_1 to the sink which passes through A_3 , and A_3 would be an origin violator.

Suppose that P contains A_2 . If $A_3 = A_2$ or A_3 comes after A_2 on P , then P contains a path from the terminal node of A_1 (the origin node of A_2) to the sink which passes through A_3 , and A_3 would be an origin violator. If A_3 comes before A_2 on P , then P contains a path from the source to the terminal node of A_1 (the origin node of A_2) which passes through A_3 , and A_3 would be a terminal violator. This completes the proof of Lemma 1.

Lemma 2: A_1 is a terminal violator of A_1 and A_2 , and A_2 is an origin violator of A_1 and A_2 .

Lemma 3: Any arc A_3 cannot be both an origin violator of A_1 and A_2 and a terminal violator of A_1 and A_2 .

Proof of Lemma 3: Suppose that an arc A_3 is both an origin violator and a terminal violator. Since A_3 is an origin violator, there exists a path from the terminal node of A_1 to the origin node of A_3 . Since A_3 is a terminal violator, there exists a path from the terminal node of A_3 to the terminal node of A_1 . The existence of these two paths, however, implies the existence of a circuit which contradicts the given acyclic structure of the network. This completes the proof of Lemma 3.

Proof of Theorem 1: Since the terminal node of A_1 cannot be a cut vertex, there exists a path P from the source to the sink which does not pass through the terminal node of A_1 . Denote the arcs on P by C_1, C_2, \dots, C_p with C_{i-1} preceding C_i on P .

Suppose that none of C_1, C_2, \dots, C_p are bridges over A_1 and A_2 . Then Lemma 1 and Lemma 3 together imply that each of C_1, C_2, \dots, C_p is either an origin violator of A_1 and A_2 or a terminal violator of A_1 and A_2 but not both. Since the origin node of C_1 is the source, C_1 cannot be an origin violator and must be a terminal violator.

Similarly, since the terminal node of C_p is the sink, C_p cannot be a terminal violator and must be an origin violator. Hence, there exists $j \geq 1$ such that C_1, C_2, \dots, C_j are all terminal violators and C_{j+1} is an origin violator.

Since C_j is a terminal violator, there exists a path from the terminal node of C_j to the terminal node of A_1 . Furthermore, since C_{j+1} is an origin violator, there is a path from the terminal node of A_1 to the origin node of C_{j+1} (the terminal node of C_j). These two paths imply the existence of a circuit from the terminal node of

C_j to the terminal node of A_1 and then back to the terminal node of C_j . (The definition of P implies that the terminal node of C_j is not the terminal node of A_1 .) This contradicts the given acyclic structure of the network and completes the proof of Theorem 1.

APPENDIX B

New Subnetwork Analysis Program

The New Subnetwork Analysis Program is an implementation of the analytical procedure described in Section 1 of this report. The basic required input is

- (a) an acyclic network with one source and one sink,
- (b) two points from each component activity's completion time distribution
- (c) probability P to be associated with the lower point,
and
- (d) specified values for the algorithm parameters θ and λ .

The output is mainly

- (a) upper and lower bounds on the moments of the network completion time, $T_r^+(\theta, \lambda)$ and $T_r^-(\theta, \lambda)$ $r = 1, 2, \dots, 10$;
- (b) upper and lower bounds on the distribution function of the network completion time, $F^+(\cdot; \theta, \lambda)$ and $F^-(\cdot; \theta, \lambda)$;
and
- (c) an approximate network completion time distribution,
$$F(\cdot; \theta, \lambda) = \frac{1}{2} [F^+(\cdot; \theta, \lambda) + F^-(\cdot; \theta, \lambda)].$$

The basic computational technique for determining critical path times is the Simplex Algorithm. This algorithm is applied to the dual problem. The Simplex Algorithm is used instead of the standard network analysis techniques because the Simplex Algorithm is ideally suited for the type of parametric programming required to evaluate several critical path times when only the activity times vary from one problem to the next.

A listing of the Subnetwork Analysis Program and a program flowchart are given at the end of this appendix.

Specific Input Instructions:

Card 1. Col. 1-3: The number of activities in the network, Format (I3).

Col. 4-6: The number of nodes in the network, Format (I3).

For each activity one card with:

Col. 11-15: The origin node of the activity, Format (I5).

Col. 21-25: The terminal node of the activity, Format (I5).

Col. 31-40: The lower point on the activity's completion time distribution, Format (F10.0).

Col. 41-50: The upper point on the activity's completion time distribution, Format (F10.0).

Col. 51-60: The probability P to be associated with the lower point (1-P will be associated with the upper point), Format (F10.5).

Next Card. Col. 1: OPTON1. OPTON1=1 implies that the program will terminate after the clusters have been formed on the basis of associates and eliminants. OPTON1#1 implies that the program will follow the normal procedure.

Col. 2: OPTON2. OPTON2=1 implies that the lower bounds on the moments of the project completion time and the upper bound on its distribution will be determined by using all activity times outside the cluster at their means instead of their lower points. This is only guaranteed to be a valid procedure when all (P, Q) = (1/2, 1/2). OPTON2#1 implies that the program will follow the normal procedure.

Next Card. Col. 1-3: IEDF. The program computes an absolute upper and lower bound for the network completion time. This range is subdivided into IEDF equal parts and the approximate distribution function (F^+ , F^- , \hat{F}) values are printed at each of these dividing points. IEDF would usually be between 10 and 100. IEDF, Format (I3).

Next Card. Col. 1-5: θ , Format (F5.2).

Col.6-10: λ , Format (F5.2).

Next Card. Col.1-10: SAMSIZ. The number of activity time configurations to be randomly selected for explicit consideration in each cluster analysis.
If $SAMSIZ < 0$ or $SAMSIZ > 2^n$, all activity time configurations will be explicitly considered - no random sampling will be done. Format (I10).

The nodes should be numbered 1, 2, ..., n with the source being number 1, the sink being number n, and the other node numbers being arbitrary. The activities should be numbered 1, 2, ... in any order desired.

Current Dimension Restrictions:

Currently the program is dimensioned for a maximum of

60 Activities

40 Nodes

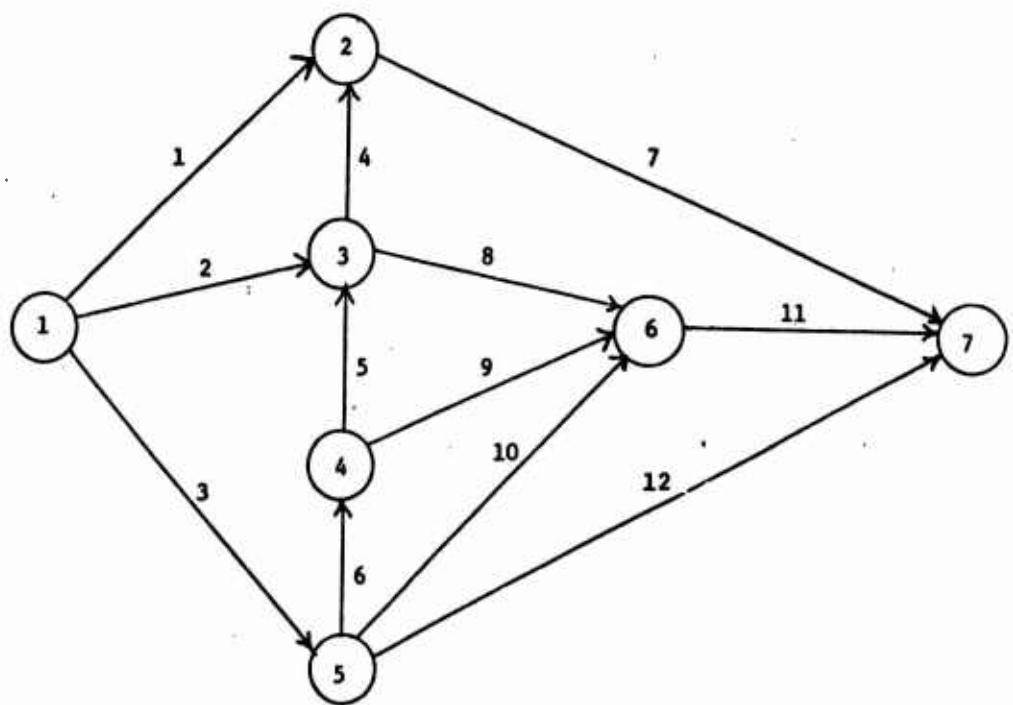
25 Clusters

25 Activities/Cluster and IEDF ≤ 100 .

Example:

The Program's input and output are illustrated in terms of the network in Figure B-1.

Figure B-1: New Subnetwork Analysis Program Example Network



SAMPLE INPUT

35

012C07

1	2	17.26	19.44	.5
1	3	19.26	21.44	.9
1	5	12.76	15.91	.3
3	2	3.51	4.01	.8
4	3	3.01	5.43	.75
5	4	3.52	4.25	.5
2	7	13.75	14.48	.65
3	6	5.05	8.43	.1
4	6	5.36	6.51	.7
5	6	8.78	11.44	.55
6	7	15.76	17.21	.5
5	7	14.32	18.35	.9

00

020

1. 1.

-1

SAMPLE OUTPUT**INITIAL INPUT**

ACTIVITY	ORIGIN	TERMINAL	LOWER POINT	UPPER POINT	MEAN	STANDARD DEVIATION	PROB. LOWER PT.
1	1	2	17.2600	19.44400	18.3500	1.0900	0.5000
2	1	3	10.2600	21.4600	10.4750	0.6545	0.9000
3	1	5	12.7600	15.0100	14.3650	1.4435	0.3000
4	3	2	3.5100	4.0120	3.6100	0.2000	0.8000
5	4	3	3.0100	5.4700	1.6150	1.0479	0.7500
6	5	4	3.5200	4.02500	3.4950	0.3650	0.5000
7	2	7	13.7500	14.4900	14.0255	0.3482	0.6500
8	3	6	5.0500	9.4300	9.0920	1.0140	0.1000
9	4	6	5.3600	6.5100	5.7050	0.5270	0.7000
10	5	6	8.7800	11.4400	9.9770	1.3233	0.5500
11	6	7	15.7600	17.2100	16.5500	0.5800	0.8000
12	5	7	14.3200	18.3500	14.7230	1.2090	0.9000

THE CRITICAL PATH TIME WHEN EACH ACTIVITY'S COMPLETION TIME IS SET EQUAL TO ITS MEAN IS = 0.46607002

THE 4 NODES ON THE CRITICAL PATH ARE AS FOLLOWS BEGINNING WITH THE TERMINAL NODE:
7. 6. 3. 4. 1.

THE 5 CRITICAL ACTIVITIES ARE AS FOLLOWS BEGINNING WITH THE TERMINAL ACTIVITY:
11. A. S. 6. 3.

THETA = 0.1000001 LAMBDA = 0.1000001

A LOWER BOUND ON THE EXPECTED CRITICAL PATH TIME IS = C.4010002

THE 6 NODES ON THE CRITICAL PATH ARE AS FOLLOWS BEGINNING WITH THE TERMINAL NODE:
7. 6. 3. 4. 5. 1.

TMF. 5 CRITICAL ACTIVITIES ARE AS FOLLOWS BEGINNING WITH THE TERMINAL ACTIVITY:
11. A. S. 6. 3.

A UPPER BOUND ON THE EXPECTED CRITICAL PATH TIME IS = 0.51233CD C?

THE 6 NODES ON THE CRITICAL PATH ARE AS FOLLOWS BEGINNING WITH THE TERMINAL NODE:
7. 6. 3. 4. 5. 1.

THE 5 CRITICAL ACTIVITIES ARE AS FOLLOWS BEGINNING WITH THE TERMINAL ACTIVITY:
11. A. S. 6. 3.

THE ASSOCIATES ARE NOW IDENTIFIED:

THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE	1-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY	11. IS = 0
THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE	2-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY	8. IS = 0
THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE	3-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY	5. IS = 0
THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE	4-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY	6. IS = 0
THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE	5-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY	3. IS = 0

THERE ARE 5 NONEMPTY CLUSTERS AFTER PULLING ON THE BASIS OF ASSOCIATES ONLY.

THE ACTIVITIES IN THE 1-TH CLUSTER ARE AS FOLLOWS:

11.

THE ACTIVITIES IN THE 2-TH CLUSTER ARE AS FOLLOWS:

9.

THE ACTIVITIES IN THE 3-TH CLUSTER ARE AS FOLLOWS:

5.

THE ACTIVITIES IN THE 4-TH CLUSTER ARE AS FOLLOWS:

6.

THE ACTIVITIES IN THE 5-TH CLUSTER ARE AS FOLLOWS:

3.

THE CLUSTERS TO WHICH EACH ACTIVITY BELONGS:

(ZERO IMPLIES THAT THE ACTIVITY IS NOT IN ANY CLUSTER)

THE 1-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 2-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 3-TH ACTIVITY IS IN THE 5-TH CLUSTER

THE 4-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 5-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 6-TH ACTIVITY IS IN THE 4-TH CLUSTER

THE 7-TH ACTIVITY IS IN THE 5-TH CLUSTER

THE 8-TH ACTIVITY IS IN THE 2-TH CLUSTER

THE 9-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 10-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 11-TH ACTIVITY IS IN THE 1-TH CLUSTER

THE 12-TH ACTIVITY IS IN THE 0-TH CLUSTER

THERE ARE 7 ACTIVITIES NOT IN ANY CLUSTER YET.

THE ELIMINANTS OF EACH NON-CRITICAL-PATH ACTIVITY ARE NOW DETERMINED:

TYPE A, i.e. ACTIVITIES NOT ON THE CRITICAL PATH. THEY ARE AS FOLLOWS:

1.	1
2.	2
3.	4
4.	7
5.	9
6.	10
7.	12

THE COMPLETION TIME FOR THE 1-TH ACTIVITY HAS BEEN CHANGED TO C.10440D 02
THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 1

THE COMPLETION TIME FOR THE 2-TH ACTIVITY HAS BEEN CHANGED TO C.20152D 02
THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 2

THE COMPLETION TIME FOR THE 4-TH ACTIVITY HAS BEEN CHANGED TO C.39100D 01
THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 4

THE COMPLETION TIME FOR THE 7-TH ACTIVITY HAS BEEN CHANGED TO C.14354D 02
THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 7

THE COMPLETION TIME FOR THE 9-TH ACTIVITY HAS BEEN CHANGED TO C.62320D 01
THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 9

THE COMPLETION TIME FOR THE 10-TH ACTIVITY HAS BEEN CHANGED TO C.11300D 02
THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 10

THE COMPLETION TIME FOR THE 12-TH ACTIVITY HAS BEEN CHANGED TO C.15032D 02
THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 12

- THERE ARE 5 CLUSTERS.
THERE ARE 1 ACTIVITIES IN THE 1-TH CLUSTER. THEY ARE AS FOLLOWS:
1.
1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:
1.
THERE ARE 1 ACTIVITIES IN THE 2-TH CLUSTER. THEY ARE AS FOLLOWS:
2.
1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:
2.
THERE ARE 1 ACTIVITIES IN THE 3-TH CLUSTER. THEY ARE AS FOLLOWS:
3.
1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:
3.
THERE ARE 1 ACTIVITIES IN THE 4-TH CLUSTER. THEY ARE AS FOLLOWS:
4.
1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:
4.
THERE ARE 1 ACTIVITIES IN THE 5-TH CLUSTER. THEY ARE AS FOLLOWS:
5.
1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:
5.
THE FOLLOWING TABLES WERE DETERMINED CONSIDERING ALL ACTIVITY CONFIGURATIONS.
THE INITIALIZATION PARAMETER FOR ANY SAMPLING IS IV = 77

A LOWER BOUND. T-(1:THETA.LAMDA). ON THE 1-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.43142D 92
A LOWER BOUND. T-(2:THETA.LAMDA). ON THE 2-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.18623D 04
A LOWER BOUND. T-(3:THETA.LAMDA). ON THE 3-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.8042AD 05
A LOWER BOUND. T-(4:THETA.LAMDA). ON THE 4-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.34752D 07
A LOWER BOUND. T-(5:THETA.LAMDA). ON THE 5-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.15023D 09
A LOWER BOUND. T-(6:THETA.LAMDA). ON THE 6-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.64968D 10
A LOWER BOUND. T-(7:THETA.LAMDA). ON THE 7-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.28108D 12
A LOWER BOUND. T-(8:THETA.LAMDA). ON THE 8-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.12165D 14
A LOWER BOUND. T-(9:THETA.LAMDA). ON THE 9-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.52667D 15
A LOWER BOUND. T-(10:THETA.LAMDA). ON THE 10-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.22809D 17

AN UPPER BOUND ON THE NETWORK COMPLETION TIME DISTRIBUTION: F+(.:THETA;LAMDA)

F+(0.40657D 02:	0.10CCCD 01:	0.10CCCD C1) E	0.10CCCD 00
F+(0.41213D C2:	C.10CCCD 01:	C.10CCCD C1) E	C.10CCCD 00
F+(0.41770D 02:	C.10CCCD 01:	C.10CCCD C1) E	C.10CCCD 00
F+(0.42326D 02:	C.10CCCD 01:	C.10CCCD C1) E	C.10CCCD 00
F+(0.42983D C2:	0.10CCCD 01:	0.10CCCD C1) E	0.10CCCD 00
F+(0.43439D 02:	0.10000D 01:	0.10000D C1) E	C.10000D 00
F+(0.43996D 02:	C.10CCCD 01:	C.10CCCD C1) E	C.10000D 00
F+(0.44552D C2:	0.10200D 01:	C.10CCCD 01) E	C.10000D 01
F+(0.451C9D 02:	0.10CCCD 01:	0.10CCCD C1) E	0.10000D 01
F+(0.45665D C2:	0.10CCCD 01:	C.10CCCD 01) E	0.10CCCD 01
F+(0.46221D 02:	0.10CC3D 01:	0.10CC3D C1) E	0.10000D 01
F+(0.46778D 02:	0.10000D 01:	C.10000D C1) E	C.10000D 01
F+(0.47334D 02:	0.10000D 01:	C.10CCCD C1) E	0.10000D 01
F+(0.47891D C2:	0.10000D 01:	C.10CCCD C1) E	0.10000D 01
F+(0.48447D 02:	0.10CCCD 01:	C.10CCCD C1) E	0.10000D 01
F+(0.49004D C2:	C.10000D 01:	C.10000D C1) E	0.10000D 01
F+(0.4955CD C2:	C.10CCCD 01:	C.10CCCD C1) E	C.10CCCD 01
F+(0.50117D 02:	0.10CCCD 01:	C.10CCCD C1) E	C.10000D 01
F+(0.50673D C2:	C.10000D 01:	C.10000D C1) E	0.10000D 01
F+(0.51230D C2:	C.10CCCD 01:	C.10CCCD C1) E	0.10000D 01

**COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION**

AN UPPER ROUND. T+(1:THFTALLAMDA). ON THE 1-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.4941SD 02
 AN UPPER BOUND. T+(2:THETA.LAMRDA). ON THE 2-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.2442SD 04
 AN UPPER ROUND. T+(3:THETA.LAMRDA). ON THE 3-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.120A3D 06
 AN UPPFR ROUND. T+(4:THETALLAMDA). ON THE 4-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.5978D C7
 AN UPPFR ROUND. T+(5:THFTALLAMDA). ON THE 5-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.296CCD 09
 AN UPPFR ROUND. T+(6:THFTALLAMRDA). ON THE 6-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.14661D 11
 AN UPPFR BOUND. T+(7:THETA.LAMRDA). ON THE 7-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.7265SD 12
 AN UPPFR ROUND. T+(8:THETA.LAMRDA). ON THE 8-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.36023D 14
 AN UPPER ROUND. T+(9:THETALLAMDA). ON THE 9-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.17870D 16
 AN UPPFR ROUND. T+(10:THFTALLAMRDA). ON THE 10-TH MOMENT OF THE NETWORK COMPLETION TIME = 0.88694D 17

A LOWER BOUND ON THE NETWORK COMPLETION TIME DISTRIBUTION: F=4; THETA(LAMUDA)

F=4	0.46657D 02:	0.10CCD 01:	0.10CCD C11 = C.10CCD C11 = C.10CCD 01
F=4	0.41213D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.4177CD 02:	C.10CCD C11 = C.10CCD 01	C.10CCD C11 = C.10CCD 01
F=4	0.42326D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.42483D 02:	C.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.43433D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.43696D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.44552D 02:	C.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.4519D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.45665D 02:	C.10CCD C11 = C.10CCD 01	C.10CCD C11 = C.10CCD 01
F=4	0.46221D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.46778D 02:	C.10CCD C11 = C.10CCD 01	C.10CCD C11 = C.10CCD 01
F=4	0.47334D 02:	C.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.47891D 02:	C.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.48447D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.490C4D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.4956CD 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.50117D 02:	C.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.50673D 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01
F=4	0.5123CD 02:	0.10CCD 01:	C.10CCD C11 = C.10CCD 01

AN APPROXIMATE NETWORK COMPLETION TIME DISTRIBUTION:

```

F(.;THETA,.LAMMDA) = .5 * ( F+(.;THETA,LAMMDA) + F-(.;THETA,LAMMDA) )

```

F(0.40657D 02;	0.10CCCC 01;	C.1CCCCD C1) =	C.50CCCC-01
F(0.41213D C2;	0.10CCCC 01;	C.10CCCC C1) =	C.5CCCCD-01
F(0.4177CD C2;	0.10CCCC 01;	C.1CCCCD C1) =	C.50CCCC-01
F(0.42326D 02;	0.1CCCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.4283D 02;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.43439D 02;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.43996D 02;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.44552D C2;	0.1CCCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.451C99 02;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.45665D 02;	0.1CCCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.46221D C2;	0.1CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.46776D C2;	0.1CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.47334D C2;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.47891D C2;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.48447D C2;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.49CC4D C2;	0.1CCCCD 01;	C.10CCCC 01;	C.50CCCC-01
F(0.49566CD C2;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.50117D C2;	0.1CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.50673D C2;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01
F(0.5123r0 02;	0.10CCCC 01;	C.10CCCC 01;	C.50CCCC-01

**COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION**

```

C
C      NEW SUBNETWORK ANALYSIS PROGRAM
C
C      IMPLICIT REAL*8 (A-H,O-Z)
C      FOR THE SAKE OF IDENTIFYING THE APPROPRIATE DIMENSIONS, LET
C          M = THE NUMBER OF ACTIVITIES IN THE NETWORK
C          NMM = NUMBER OF NODES IN THE NETWORK
C          NMMP1 = NMM + 1
C          N = M + NMM
C          L = THE LENGTH OF THE CRITICAL PATH
C          C = THE MAXIMUM NUMBER OF BRANCHES IN A CLUSTER
C          IEDF = THE NUMBER OF DIVISIONS IN THE EMPIRICAL
C                  DISTRIBUTION FUNCTION
C
C
C      INTEGER TAIL( M ),HEAD( M ),ASSGRP( L,L ),CLINCL(L,L ),EGRP(L)
C      DIMENSION NINCL( C ),INCLUS( L, C ),NCLINC( L )
C      DIMENSION FD(IEDF),NLEFD(L,IEDF),NSAVE(IEDF),NIR(L)
C      DIMENSION AVG( L ),THAT(L)
C      DIMENSION LEFT( M ),LEFTO(M ),NONCP(M)
C      DIMENSION INRAISE(NMM),XNODE(NMM)
C      DIMENSION XB1(NMMP1),Y1(NMMP1),REDCOS(N),ISTAT(N)
C      DIMENSION ICRTIP(L),NINAG(M),ICRITN(L+1),CTIME(N),COT(N)
C      DIMENSION K3(L),IBH(L),FLO(M),FHI(M),SIGMA(M),B1INV(NMMP1,NMMP1)
C      REAL        MOMENT(L,10)
C      DIMENSION PP(M),PQ(M)
C
C      OF COURSE THESE DIMENSIONS ARE MERELY UPPER BOUNDS
C
C
C      COMMON B1INV,REDCOS,CTIME,XB1,INBASE,HEAD,TAIL,NMMP1,NMM,N,ISTAT
C      COMMON N,MP1
C      INTEGER TAIL(60),HEAD(60),ASSGRP(25,25),CLINCL(25,25),EGRP(25)
C      INTEGER SAMSIZ,RANSAM
C      DIMENSION NTNCL(25),INCLUS(25,25),NCLINC(25)
C      DIMENSION FD(100),NTB(25)
C      REAL*8 NLEFD(25,100),NSAVE(100)
C      DIMENSION AVG(25),THAT(25)
C      DIMENSION INBASE(40)
C      DIMENSION XNODE(40)
C      DIMENSION XB1(41),Y1(41),REDCOS(100),ISTAT(100)
C      DIMENSION B1INV(41,41),KB(25),TBB(25),FLO(60),FHI(60),SIGMA(60)
C      DIMENSION ICRTIP(25),NINAG(50),ICRITN(26),COT(100),CTIME(100)
C      DIMENSION LEFT(60),LEFTO(60),NONCP(60)
C      REAL*8 LAMBDA,MOMENT(25,10)
C      DIMENSION PP(60),PQ(60)
C      INTEGER OPTON1,OPTON2
C          M = THE NUMBER OF ACTIVITIES IN THE NETWORK
C          NMM = THE NUMBER OF NODES IN THE PERT NETWORK
C
100     READ(5,100)  M,NMM
C      FORMAT(2I3)
C      N=NMM+M
C      MP1=M+1
C      NMMP1=NMM+1
C
C      THE ACTIVITIES ARE DESCRIBED IN TERMS OF THEIR NODES
C      II=THE TAIL NODE, THE ORIGIN NODE
C      JJ=THE HEAD NODE, THE TERMINAL NODE
C      FLO = THE LOWER POINT

```

```

C      FHI = THE UPPER POINT          60
C      SIGMA = (FHI - FLO)*DSORT( PP*(1-PP)) = STD. DEVIATION 61
C      PP = THE PROBABILITY OF THE LOWER POINT 62
C
C      DO 610 I=1,M 63
C        READ(5,2501) I,JJ,FLO(I),FHI(I),PP(I)
C        PQ(I)=1.00-PP(I)
C        SIGMA(I)=(FHI(I)-FLO(I))*DSORT( PP(I)*PQ(I))
2501   FORMAT(10X,I5.5X,I5.5X,F10.0,F10.0,F10.5) 64
C        COT(I)=PP(I)*FLO(I)+PQ(I)*FHI(I) 65
C        CTIME(I) = COT(I) 66
C        TAIL(I) = I 67
C
C        610 HEAD(I)=JJ 68
C
C        COT = THE ORIGINAL RIGHT-HAND SIDES, I.E. THE MEANS 69
C          OF FLO AND FHI 70
C
C        CTIME = THE CURRENT RIGHT-HAND SIDES 71
C
C        DO 55610 I=MP1,N 72
C55610 CTIME(I) = 0. 73
C
C        OPTION1 =1 IMPLIES THAT THE PROGRAM WILL TERMINATE AFTER 74
C          THE CLUSTERS HAVE BEEN FORMED. NO BOUNDS ON 75
C          THE PROJECT COMPLETION TIME MOMENTS OR 76
C          DISTRIBUTION WILL BE DETERMINED. 77
C
C        OPTION1 NOT= 1 IMPLIES THAT THE NORMAL PROCEDURE WILL BE 78
C          FOLLOWED. 79
C
C        OPTION2 =1 IMPLIES THAT THE LOWER BOUNDS ON THE MOMENTS 80
C          AND THE UPPER BOUND ON THE DISTRIBUTION WILL 81
C          BE DETERMINED USING ALL ACTIVITY TIMES OUTSIDE 82
C          THE CLUSTER AT THEIR MEAN. THIS PROCEDURE IS 83
C          ONLY GUARANTEED TO BE VALID WHEN ALL 84
C          (P,Q) = (.5,.5). 85
C
C        OPTION2 NOT= 1 IMPLIES THAT THE NORMAL PROCEDURE WILL BE 86
C          FOLLOWED. 87
C
C
C        READ(5,77551) OPTION1,OPTION2 88
C77551 FORMAT(10I1)
C        I=OPTION1+OPTION2 89
C        IF(I.GE.1) WRITE(6,77553) 90
C
C77553 FORMAT(1H1)
C        IF(OPTION1 .EQ.1) WRITE(6,77552) 91
C77552 FORMAT(1H0,10X,'OPTION1=i AND THE PROGRAM WILL TERMINATE AFTER THE 92
C          * CLUSTERS HAVE BEEN FORMED.',/,1IX,'NO BOUNDS ON THE PROJECT COMPL 93
C          *ETION TIME MOMENTS OR DISTRIBUTION WILL BE DETERMINED.')
C        IF(OPTION2 .EQ.1) WRITE(6,77554) 94
C77554 FORMAT(1H0,10X,'OPTION2=1 AND THE LOWER BOUNDS ON THE PROJECT COMP 95
C          *ETION TIME MOMENTS AND THE UPPER BOUND ON THE PROJECT COMPLETION T 96
C          *IME DISTRIBUTION',/,1IX,'WILL BE DETERMINED USING ALL ACTIVITY TIM 97
C          *ES OUTSIDE THE CLUSTER AT THEIR MEAN.',/,1IX,'THIS PROCEDURE IS ON 98
C          *LY GUARANTEED TO BE VALID WHEN ALL (P,Q) = (.5,.5) .')
C
C        IEDF = THE NUMBER OF DIVISIONS IN THE EMPIRICAL 99
C          DISTRIBUTION FUNCTION 100
C
C        READ(5,1001) IEDF 101
C        WRITE(6,2700)
C
C2700   FORMAT(1H1,15X,'INITIAL INPUT') 102
C        WRITE(6,2701)
C
C2701   FORMAT(1H0,10X,'ACTIVITY ORIGIN TERMINAL LOWER POINT UPPER P 103
C          *OINT MEAN STANDARD DEVIATION PROB. LOWER PT.') 104
C
C        DO 2704 I=1,M 105
C2704   WRITE(6,2702) I,TAIL(I),HEAD(I),FLO(I),FHI(I),CTIME(I),SIGMA(I),PP 106
C          *(I) 107

```

2702 FORMAT(1H ,13X,13.5X,13.7X,13.5X,F10.4,3X,F10.4,3X,F10.4,4.4X,F10.4,
*14X,F6.4)

C
C

THE FOLLOWING INDICATORS ARE USED:

C
C
IPARM = 1 IMPLIES THE CRITICAL PATH TIME WHEN ALL ACTIVITY
COMPLETION TIMES ARE SET EQUAL TO THEIR MEANS IS
BEING DETERMINED

C
C
IPARM = 2 IMPLIES THAT THE LOWER BOUND ON THE COMPLETION TIME
FOR THE SUBNETWORK IS BEING DETERMINED

C
C
IPARM = 3 IMPLIES THAT THE UPPER BOUND ON THE COMPLETION TIME
FOR THE SUBNETWORK IS BEING DETERMINED

C
C
IPARM > 3 WHEN INDEXL=0 IMPLIES THAT THE ASSOCIATES ARE
BEING DETERMINED

C
C
INDEXL=0 IMPLIES THAT INITIAL CLUSTERS ARE STILL BEING FORMED

C
C
INDEXL=1 IMPLIES THAT THE LEFTOVERS, THEIR ELIMINANTS, AND

POOLED CLUSTERS ARE BEING DETERMINED

C
C
INDEXL=2 IMPLIES THAT THE 2**NINCL() RUNS FOR EACH CLUSTER
ARE BEING MADE

C
C
ICBCP = 0 IMPLIES THAT THE PROCEDURE FOR DETERMINING UPPER
BOUNDS ON THE MOMENTS OF THE NETWORK COMPLETION TIME
AND LOWER BOUNDS ON THE DISTRIBUTION OF COMPLETION
TIMES HAS NOT BEEN REBEGUN

C
C
ICBCP = 1 IMPLIES THAT THE PROCEDURE FOR DETERMINING UPPER
BOUNDS ON THE MOMENTS OF THE NETWORK COMPLETION TIME
AND LOWER BOUNDS ON THE DISTRIBUTION OF COMPLETION
TIMES IS BEING INITIALIZED

C
CONTINUE

C
C
ICBCP = 2 IMPLIES THAT THE PROCEDURE FOR DETERMINING UPPER
BOUNDS ON THE MOMENTS OF THE NETWORK COMPLETION TIME
AND LOWER BOUNDS ON THE DISTRIBUTION OF COMPLETION
TIMES IS BEING CARRIED OUT

IPARM=1
INDEXL=0
ICBCP=0
601C CONTINUE
DO 104 I=1,NMM

104 INPASF(I)=M+I
DO 2001 J=1,M

2001 ISTAT(J)=0.
DO 2002 J=MPI,N

2002 ISTAT(J)=1
DO 10 II=1,NMMP1
DO 12 L=1,NMMP1

12 B1INV(L,II) = 0.

10 B1INV(II,II) = 1.
DO 30 I=1,NMM

30 XB1(I) = 0.
XB1(NMMP1) = 1.

TOLR1=1.0D-10

*COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION*

120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179

```

C      START THE SIMPLFX ALGORITHM          180
C      SOLVE THE DUAL PROBLEM               181
C      THE NUMBER OF VARIABLES IS M REAL + NMM SLACKS 182
C      FOR A TOTAL OF N VARIABLES          183
C
350    CONTINUE                           184
280C DO 23 J=1,N                         185
      RATS = 0.                            186
      IF (ISTAT(J).EQ.1) GO TO 52800     187
      IF (J.GT.M) GO TO 22                188
      RATS = -B1INV(1,HEAD(J)+1)+B1INV(1,TAIL(J)+1) + CTIME(J)
      GO TO 52800                         189
22    RATS = -B1INV(1,J-M+1)              190
52800 RFDCCS(J)= RATS                  191
23    CONTINUE                           192
22800 CONTINUE                           193
      IRMAX=1                            194
      RMAX=RFDCCS(1)                     195
      DO 24 J=2,N                         196
      IF(RFDCCS(J) .LE. RMAX) GO TO 24   197
      RMAX=PEDCCS(J)                     198
      IRMAX=J                            199
24    CONTINUE                           200
      IF(RMAX .LE. TOLR1) GO TO 401     201
22824 CONTINUE                           202
      DO 26 L=1,NMMP1                   203
      IF (IRMAX.GT.M) GO TO 50026       204
      Y1(L) = -B1INV(L,TAIL(IRMAX)+1)+B1INV(L,HEAD(IRMAX)+1)
      GO TO 26                           205
50026 Y1(L) = B1INV(L,IRMAX-M+1)        206
26    CONTINUE                           207
      Y1(1) = Y1(1) - CTIME(IRMAX)      208
      NUMBER=2                            209
      DO 27 L=2,NMMP1                   210
      IF(Y1(L) .LE. TOLR1) NUMBER=NUMBER+1
      IF(NUMBER .EQ. NMM) GO TO 403     211
      RMIN=.99D 20
      IRMIN=0.                            212
      DO 32 II=2,NMMP1                   213
      IF(Y1(II).LE. TOLR1) GO TO 32     214
      RATS =XB1(II)/Y1(II)               215
      RR=RATS-RMIN                      216
      IF(RR .GE. 0.D0) GO TO 32         217
      RMIN=RATS                          218
      IRMIN=II                           219
32    CONTINUE                           220
      DO 33 J=2,NMMP1                   221
      WW=B1INV(IRMIN ,J)/Y1(IRMIN )    222
      DO 37 L=1,NMMP1                   223
      B1INV(L ,J)=B1INV(L,J)-WW*Y1(L)  224
37    B1INV(IRMIN ,J)=WW              225
33
C      UPDATE THE BASIC VARIABLES: INBASE AND XB1 226
C
      ISTAT(INBASE(IPMIN-1))=0          227
      ISTAT(IRMAX)=1                    228
      INBASF(IRMIN-1)=IRMAX            229
      W=XB1(IRMIN )/Y1(IRMIN )
      DO 38 I=1,NMMP1                  230
      XB1(I)=XB1(I)-Y1(I)*W           231
38

```

```

XB1(IRMIN)=N
GO TO 350
240
403 WRITE(6,530)
241
530 FORMAT(1H0.5X,'NO FEASIBLE SOLUTION EXISTS. CHECK YOUR INPUT DATA
*,')
242
*.,'
243
WRITE(6,250)
244
850 FORMAT(1H1)
245
GO TO 999
246
C
247
C      END OF THE SIMPLEX ALGORITHM
248
C
250
401 CONTINUE
251
IF(ICRCP.EQ.1) GO TO 6008
252
IF(INDEXL.EQ.2) GO TO 3204
253
C
254
C      KKK= THE NUMBER OF NODES ON THE CRITICAL PATH
255
C      KB(L)= THE L-TH NODE IN THE CRITICAL PATH, COUNTING BACKWARDS
256
C          FROM THE TERMINAL NODE
257
C      KKB= THE NUMBER OF ACTIVITIES ON THE CRITICAL PATH
258
C      TBB(L)= THE L-TH ACTIVITY ON THE CRITICAL PATH, COUNTING
259
C          BACKWARDS FROM THE TERMINAL NODE
260
C
261
CONTINUE
262
C
263
C      INBASE IS A SET OF M INTEGER VARIABLES WHICH INDICATE THE
264
C      COMPOSITION OF THE CURRENT BASIS. FOR EXAMPLE,
265
C      INBASE(K) = 7 IMPLIES THAT THE K-TH COLUMN IN THE BASIS B
266
C          CORRESPONDS TO THE 7-TH VARIABLE
267
C
268
C
269
C      ISTAT INDICATES THE BASIC STATUS OF EACH VARIABLE
270
C      ISTAT(K) = 1 IMPLIES THAT THE K-TH VARIABLE IS IN THE
271
C          DUAL BASIS
272
C      ISTAT(K) = 0 IMPLIES THAT THE K-TH VARIABLE IS NOT IN THE
273
C          DUAL BASIS
274
C
275
C
276
C      THE FOLLOWING STATEMENTS DETERMINE THE NODES AND ACTIVITIES ON
277
C      THE CRITICAL PATH
278
C
279
C
280
C      THE DUAL SOLUTION IMPLIES THE FOLLOWING OPTIMAL SOLUTION TO THE
281
C      PRIMAL PERT PROBLEM. HOWEVER SOME OF THE NODE TIMES(OTHER THAN
282
C      THE LAST ONE) MAY BE HIGHER THAN NECESSARY. THUS IN
283
C      DETERMINING THE CRITICAL PATH AN ALTERNATIVE OPTIMAL SOLUTION
284
C      MAY HAVE TO BE IDENTIFIED.
285
C      RIINV IS NOT CHANGED.
286
C
287
DO B3002 I=1,NMM
288
83002 XNCDE(I)=B1INV(I,I+1)
289
KKK=1
290
KB(I)=NMM
291
83001 IK=KB(KKK)
292
C
293
C      DETERMINE WHETHER THE TIME TO REACH NODE IK IS NECESSARILY
294
C      AS LARGE AS INDICATED FROM THE DUAL SOLUTION
295
C
296
SMIN=999999.
297
ISMIN=0
298
DO B3000 I=1,M
299

```

```

IF(HEAD(I).NE.IK) GO TO 83000          300
SI ACK=XNODE(HEAD(I))-XNODE(TAIL(I))-CTIME(I)
IF(SLACK.GE.SMIN) GO TO 83000          301
SMIN=SLACK                           302
ISMIN=I                             303
83000 CONTINUE                         304
IF(SMIN.LT.0.0001) GO TO 83003          305
C                                         306
C             THE TIME FOR NODE IK WAS UNNECESSARILY LARGE      307
C
XNODE(IK)=XNODE(IK)-SMIN              308
KKK=KKK-1                            309
GO TO 83001                          310
83003 IRR(KKK)=ISMIN                311
KKK=KKK+1                            312
KP(KKK)=TAIL(ISMIN)                  313
IF(TAIL(ISMIN).GT.1) GO TO 83001      314
KKR=KKK-1                            315
IF(INDEXL.EQ.1) GO TO 3121           316
IPARM=IPARM+1                        317
IF(IPARM.GT.4) GO TO 2910            318
IF(IPARM.EQ.3) GO TO 6400            319
IF(IPARM.EQ.4) GO TO 6401            320
C                                         321
C             ICRITP(L)= THE L-TH ACTIVITY ON THE ORIGINAL CRITICAL PATH 322
C             KCPB= THE NUMBER OF ACTIVITIES ON THE ORIGINAL CRITICAL PATH 323
C
TOTAL = RIINV(1,NMMP1)                 324
KCPB=KKR                            325
ICRITN(1) = NMM                      326
DO 2802 I=1,KCPA                     327
ICRITN(I+1) = KB(I+1)                 328
2802 ICRITP(I)=IRR(I)                 329
X=TOTAL                            330
WRITE(6,851) X                       331
851 FORMAT(1H0,5X,'THE CRITICAL PATH TIME WHEN EACH ACTIVITY''S COMPLE 332
*TION TIME IS SET EQUAL TO ITS MFAN IS = ',D15.5)               333
WRITE(6,7626) KKK                     334
7606 FORMAT(1H0,10X,'THE ',I3,' NODES ON THE CRITICAL PATH ARE AS FOLLO 335
*WS BEGINNING WITH THE TERMINAL NODE:')                   336
WRITE(6,7707) (KB(I),I=1,KKK)         337
7707 FORMAT(15X,20(I3,','))           338
WRITE(6,7711) KKB                     339
7710 FORMAT(1H0,10X,'THE ',I3,' CRITICAL ACTIVITIES ARE AS FOLLOWS REGI 340
*NNING WITH THE TERMINAL ACTIVITY:')                   341
WRITE(6,7707) (IRR(I),I=1,KKR)        342
READ(5,2920) THETA,LAMBDA            343
2920 FORMAT(2F5.2)                   344
WRITE(6,3071) THETA,LAMBDA           345
3071 FORMAT(1H0,10X,'THETA = ',E15.5,'    LAMBDA = ',E15.5)       346
C                                         347
C             SAMSIZ = THE NUMBER OF ACTIVITY TIME CONFIGURATIONS TO BE 348
C             RANDOMLY SELECTED FOR CONSIDERATION IN EACH CLUSTER        349
C
NOTE: SINCE THIS IS A RANDOM SAMPLE . SOME PERCENTILE               350
COMBINATIONS MAY BE CONSIDERED MORE THAN ONCE.                      351
C                                         352
C             READ (5,3209) SAMSIZ          353
3209 FORMAT (I10)                   354
C                                         355

```

C THE COMPLETION TIME FOR ALL ACTIVITIES IS SET TO THEIR LOWER 360
C PERCENTILE. THE RESULTING CRITICAL PATH TIME IS A LOWER 361
C BOUND ON THE EXPECTED CRITICAL PATH TIME. 362
C
C DO 6402 I=1,M 363
6402 CTIME(I) = FLO(I) 364
CALL BINVA(62800) 365
6400 CPLB= B1INV(1,NMMP1) 366
WRITE(6,6405) CPLB 367
6405 FORMAT(1H0,5X,'A LOWER BOUND ON THE EXPECTED CRITICAL PATH TIME IS 368
* = ',F15.5) 369
WRITE(6,7606) KKK 370
WRITE(6,7707) (KB(I),I=1,KKK) 371
WRITE(6,7710) KKB 372
WRITE(6,7707) (IRB(I),I=1,KKB) 373
C
C THE COMPLETION TIME FOR ALL ACTIVITIES IS SET TO THEIR UPPER 374
C PERCENTILE. THE RESULTING CRITICAL PATH TIME IS A UPPER 375
C BOUND ON THE EXPECTED CRITICAL PATH TIME. 376
C
C DO 6406 I=1,M 377
6406 CTIMF(I) = FHI(I) 378
CALL BINVA(62800) 379
6401 CPUB= B1INV(1,NMMP1) 380
WRITE(6,6409) CPUB 381
6409 FORMAT(1H0,5X,'A UPPER BOUND ON THE EXPECTED CRITICAL PATH TIME IS 382
* = ',E15.5) 383
WRITE(6,7606) KKK 384
WRITE(6,7707) (KB(I),I=1,KKK) 385
WRITE(6,7710) KKB 386
WRITE(6,7707) (IRB(I),I=1,KKB) 387
C
C FD(I) = THE LOWER BOUND ON THE EXPECTED CRITICAL PATH TIME 388
C PLUS 1/IEDF OF THE DISTANCE TO THE UPPER BOUND 389
C NLEFD(IR,I) = THE SUM OF (THE CRITICAL PATH TIME FOR A 390
C CONFIGURATION * THE PROBABILITY OF THE 391
C CONFIGURATION --- WHEN THE CRITICAL PATH TIME IS 392
C <= FD(I)) * (THE NUMBER OF POSSIBLE 393
C CONFIGURATIONS) / (THE SAMPLE SIZE) 394
C FOR THE IR-TH CLUSTER 395
C
C FD AND NLEFD ARE USED TO BUILD AN 'EMPIRICAL' DISTRIBUTION OF 396
C THE CRITICAL PATH TIMES 397
C
C C=(CPUB-CPLB)/IEDF 398
DO 6412 K=1,KCPR 399
DO 6412 I=1,IEDF 400
FD(I)=CPLB+I*C 401
6412 NLEFD(K,I)=0.D0 402
C
C THE ASSOCIATE GROUPS ARE NOW FORMED 403
C
C WRITE(6,3165) 404
3165 FORMAT(1H1,5X,'THE ASSOCIATES ARE NOW IDENTIFIED:') 405
11111=1 406
DO 2825 I=1,M 407
2825 CTIMF(I)=COT(I) 408
1WWQ=ICRITP(I) 409
CHANG= LAMBDA*SIGMA(IWWQ) 410
TFX=COT(IWWQ)-CHANG 411

```

IF(TFX.LT.0.0) CHANG=COT(IWWWQ)
CTIME(IWWWQ)=COT(IWWWQ)-CHANG
CALL BINVA(62800)
2801 CONTINUE
IF(ISTAT(IWWWQ).EQ.1) CALL BINVI(622625,COT(IWWWQ),CTIME(IWWWQ),
*IWWWQ)
REDCOS(IWWWQ)=REDCOS(IWWWQ)+COT(IWWWQ)-CTIME(IWWWQ)
22825 CTIMF(IWWWQ)=COT(IWWWQ)
IWWWQ=ICRITP(IIII)
CHANG= LAMBDA*SIGMA(IWWWQ)
TFX=COT(IWWWQ)-CHANG
IF(TFX.LT.0.0) CHANG=COT(IWWWQ)
CTIME(IWWWQ)=COT(IWWWQ)-CHANG
IF (ISTAT(IWWWQ).EQ.1) CALL BINVI(622900,CTIME(IWWWQ),COT(IWWWQ),
*IWWWQ)
REDCOS(IWWWQ)=REDCOS(IWWWQ)-COT(IWWWQ)+CTIME(IWWWQ)
GO TO 22800
C
C DETERMINING ASSOCIATE GROUP
C
2910 NINAG(IIII)=0
DO 2911 K=1,KKR
KK=1
2913 IF(IRR(K).EQ.ICRITP(KK)) GO TO 2911
IF(KK.GE.KCPB) GO TO 2912
KK=KK+1
GO TO 2913
2912 NINAG(IIII)=NINAG(IIII)+1
ASSGRP(IIII,NINAG(IIII))=IRR(K)
2911 CONTINUE
WRITE(6,2915) IIII,ICRITP(IIII),NINAG(IIII)
2915 FORMAT(1H0,10X,'THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE ',I3,
*'-TH CRITICAL PATH ACTIVITY, I.E. ACTIVITY ',I3,', IS = ',I3)
IDUCK=NINAG(IIII)
IF(IDUCK.EQ.0) GO TO 2810
WRITE(6,2916) (ASSGRP(I)III,I),I=1,1DUCK)
2916 FORMAT(1H0,15X,'THE ACTIVITIES IN THE ASSOCIATE GROUP ARE AS FOLLO
*WS*,/,15X,50(I3,0,0))
2810 IIII=IIII+1
IF(IIII.LF.KCPB) GO TO 2801
C
C DETERMINE THE CLUSTERS
C
THE CLUSTERS ARE POOLED TOWARD THE TERMINAL NODE
NCLUS = THE NUMBER OF NON-EMPTY CLUSTERS
NINCL(I) = THE NUMBER OF ACTIVITIES IN THE I-TH CLUSTER
INCLUS(I,J) = THE J-TH ACTIVITY IN THE I-TH CLUSTER
NCLINC(I) = THE NUMBER OF CLUSTERS COMPRISING THE I-TH
CLUSTER AFTER POOLING
CLINCL(I,J) = THE J-TH CLUSTER WHICH HAS BEEN POOLED INTO
THE I-TH CLUSTER
NCLINC AND CLINCL HELP KEEP TRACK OF WHICH CLUSTER THE
CRITICAL PATH ACTIVITIES ARE IN
C
C BELOW FORMS CLUSTERS BY PUTTING EACH CRITICAL PATH ACTIVITY IN
SEPARATE CLUSTER AND THEN ADDING EACH CRITICAL PATH ACTIVITY'S
ASSOCIATES TO ITS CLUSTER
C

```

```

NCLUS=KCPB          480
DO 3020 I=1,KCPB   481
NCLINC(I)=1         482
CLINCL(I,I)=I       483
NINCL(I) =NINAG(I)+1 484
INCLUS(I,I)=ICRITP(I) 485
IF(NINAG(I).EQ.0) GO TO 3020 486
IDUCK=NINCL(I)      487
DO 3021 J=2, IDUCK 488
JJ=J-1              489
3021 INCLUS(I,J)=ASSGRP(I,JJ) 490
3020 CONTINUE        491
C                   492
C             BELOW POOLS CLUSTERS FORMED FROM ASSOCIATES 493
C                   494
IA=0                495
3031 IA=IA+1         496
IF(IA.GE.KCPB) GO TO 3030 497
IF(NCLUS.EQ.1) GO TO 3030 498
IDIA=NINCL(IA)        499
IF(IDIA.EQ.0) GO TO 3031 500
IAA=IA+1             501
DO 3023 II=IAA,KCPB 502
IDII=NINCL(II)        503
IF(IDII.EQ.0) GO TO 3023 504
DO 3025 I=1, IDIA    505
DO 3025 J=1, IDII    506
IF(INCLUS(II,J).EQ. INCLUS(IA,I)) GO TO 3027 507
3025 CONTINUE        508
GO TO 3023           509
3027 NCLUS=NCLUS-1   510
DO 3028 J=1, IDII    511
DO 3029 I=1, IDIA    512
IF(INCLUS(II,J).EQ. INCLUS(IA,I)) GO TO 3028 513
3029 CONTINUE        514
NINCL(IA)=NINCL(IA)+1 515
TNCLUS(IA,NINCL(TA)) =INCLUS(II,J) 516
3028 CONTINUE        517
NINCL(II)=0            518
NCLINC(IA)=NCLINC(IA)+1 519
CLINCL(IA,NCLINC(IA)) = II 520
NCLINC(II)=0            521
3023 CONTINUE        522
GO TO 3031           523
3030 CONTINUE        524
C                   525
C             BELOW DESCRIBES CLUSTERS AFTER POOLING BASED ON THE ASSOCIATES 526
C                   527
WRITE(6,3033) NCLUS 528
3033 FORMAT(1H1,10X,'THERE ARE ',I3,' NONEMPTY CLUSTERS AFTER POOLING O 529
*N THE BASIS OF ASSOCIATES ONLY')
II=0                530
DO 3034 I=1,KCPB   531
IF(NINCL(I).EQ.0) GO TO 3034 532
II=II+1             533
IDUCJ=NINCL(I)      534
WRITE(6,3035) I,(INCLUS(I,J),J=1, IDUCJ) 535
3035 FORMAT(1H2,10X,'THE ACTIVITIES IN THE ',I3,'-TH CLUSTER ARE AS FOL 536
*LOWS:',/,15X,50(I3,'')) 537
*3034 CONTINUE        538
3034 CONTINUE        539

```

C
C DESCRIBES WHERE EACH ACTIVITY IS BEFORE ELIMINANTS ARE
C CONSIDERED
C
C EXAMINING EACH ACTIVITY AND DETERMINE WHICH CLUSTER, IF ANY, IT IS
C IN.
C LEFT(I) = 0 IMPLIES THAT THE I-TH ACTIVITY IS NOT IN ANY
C CLUSTER
C LEFT(I) = J IMPLIES THE I-TH ACTIVITY IS IN THE J-TH
C CLUSTER
C
C WRITE(6,3104)
3104 FORMAT(1H0,10X,'THE CLUSTER TO WHICH EACH ACTIVITY BELONGS:',/,15X
* ,*(ZERO IMPLIES THAT THE ACTIVITY IS NOT IN ANY CLUSTER))
DO 3101 I=1,M
LEFT(I)=0
DO 3102 J=1,KCPB
IF(NINCL(J),EQ,0) GO TO 3102
IDUCK=NINCL(J)
DO 3110 K=1,1DUCK
IF(I,EQ,INCLUS(J,K)) GO TO 3107
3110 CONTINUE
3102 CONTINUE
GO TO 3101
3107 LFFT(I)=J
3101 WRITE(6,3103) I,LFFT(I)
3103 FORMAT(1H ,15X,'THE ',I3,'-TH ACTIVITY IS IN THE ',I3,'-TH CLUSTER
*)
INDEXL=1
C
C LEFTOVERS ARE ACTIVITIES NOT IN CLUSTERS AFTER ASSOCIATES HAVE
C BEEN CONSIDERED BUT BEFORE ELIMINANTS HAVE BEEN CONSIDERED
C
C DETERMINE THE NUMBER OF LEFTOVERS. NLEFT
C LEFTO(L) = J IMPLIES THAT THE L-TH LEFTOVER IS THE J-TH
C ACTIVITY
C
NLEFT=0
DO 3122 J=1,M
IF(LFFT(J),NE,0) GO TO 3122
NLEFT=NLEFT+1
LFFT(NLEFT)=J
3122 CONTINUE
WRITE(6,3123) NLEFT
3123 FORMAT(1H0,10X,'THERE ARE ',I3,' ACTIVITIES NOT IN ANY CLUSTER YE
*T.')
WRITE(6,3323)
3323 FORMAT(1H1,5X,'THE ELIMINANTS OF EACH NON-CRITICAL-PATH ACTIVITY A
*RE NOW DETERMINED:')
C
C ELIMINANTS FOR EACH NON-CRITICAL-PATH ACTIVITY ARE NOW
C DETERMINED
C NNCNP = THE NUMBER OF ACTIVITIES NOT ON THE CRITICAL PATH
C NCNP(LE) = THE LE-TH ACTIVITY NOT ON THE CRITICAL PATH
C
NNCNP=M-KCPB
LF=0
DO 5000 I=1,M

```

J=1          600
5001 IF(I.EQ.ICRITP(J)) GO TO 5000 601
J=J+1        602
IF(J.LE.KCPB) GO TO 5001 603
5002 LE=LE+1 604
NONCP(LE)=I 605
5003 CONTINUF 606
      WRITE(6,5005) NNNCP 607
5005 FORMAT(1H0,5X,'THERE ARE ',I3,' ACTIVITIES NOT ON THE CRITICAL PA 608
*TH. THEY ARE AS FOLLOWS:') 609
IF(NNNCP.EQ.0) GO TO 3124 610
DO 5006 I=1,LE 611
5006 WRITE(6,5007) I,NONCP(I) 612
5007 FORMAT(1H ,15X,I3,'. ',I3) 613
IF(NNNCP.EQ.0) GO TO 3124 614
LF=0          615
3126 LE=LE+1 616
IF (ISTAT(IWWWO).EQ.1) CALL BINVI(&23127,COT(IWWWO),CTIME(IWWWO), 617
*IWWWO)
      RFDCOS(IWWWO) = RFDCOS(IWWWO)-CTIME(IWWWO)+COT(IWWWO) 618
23127 CONTINUF 619
      CTIME(IWWWO) = COT(IWWWO) 620
      CTIME(NONCP(LE)) = COT(NONCP(LE)) + THFTA*SIGMA(NONCP(LE)) 621
      IF (ISTAT(NONCP(LE)).EQ.1) CALL BINVI(&7756,CTIME(NONCP(LE)), 622
*      COT(NONCP(LE)),NONCP(LE)) 623
      RFDCOS(NONCP(LE))=RFDCOS(NONCP(LE))-COT(NONCP(LF))+CTIME(NONCP(LE)) 624
*)
7756 IWWWO = NONCP(LE) 625
      WRITE(6,3152) NONCP(LF),CTIME(NONCP(LE)) 626
3152 FORMAT(1H0,///, 5X,'THE COMPLETION TIME FOR THE ',I3,'-TH ACTIVITY 627
* HAS BEEN CHANGED TO ',E15.5) 628
      GO TO 22800 629
3121 CONTINUE 630
C
C      DETERMINE THE ELIMINANTS OF THE LE-TH ACTIVITY NOT ON THE 631
C      CRITICAL PATH 632
C      NE      = THE NUMBER OF ELIMINANTS FOR THE LE-TH 633
C                  ACTIVITY NOT ON THE CRITICAL PATH 634
C      EGRP(J)    = THE J-TH ELIMINANT FOR THE LE-TH ACTIVITY 635
C                  NOT ON THE CRITICAL PATH 636
C
C      NE=0          637
DO 3130 K=1,KCPB 638
DO 3131 I=1,KKB 639
IF(ICRP(I).EQ.ICRITP(K)) GO TO 3130 640
3131 CONTINUF 641
NE=NE+1          642
EGRP(NE)=ICRITP(K) 643
3130 CONTINUE 644
      WRITE(6,3133) NE,NONCP(LE) 645
3133 FORMAT(1H0,10X,'THERE ARE ',I3,' ELIMINANTS CORRESPONDING TO ACTIV 646
*ITY ',I3) 647
      IF(NE.EQ.0) GO TO 3171 648
      DO 3135 K=1,NE 649
3135 WRITE(6,3136) K,NONCP(LE),EGRP(K) 650
3136 FORMAT(1H ,14X,'THE ',I3,'-TH ELIMINANT CORRESPONDING TO ACTIVITY 651
* ',I3,' IS ACTIVITY ',I3) 652
C
C      DETERMINE WHETHER NONCP(LE) IS AN ASSOCIATE 653
C      JA = 1  IF NONCP(LE) IS AN ASSOCIATE 654

```

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

```

C           JA = 2  IF NONCP(LE)  IS NOT AN ASSOCIATE      660
C
C           K=NCNCP(LE)                                661
C           JA=1                                662
C           IF(LFFT(K).EQ.0) JA=2                  663
C           IF(JA.EQ.2) GO TO 5010                664
C           TT=LFFT(K)                            665
C
C           THE IT-TH CLUSTER IS EXPANDED TO INCLUDE ELIMINANTS 666
C
C           GO TO 5011                                667
C
5010  CONTINUE                                668
C
C           ITTT IS THE ACTIVITY NUMBER OF THE FIRST ELIMINANT 669
C           IT IS THE CLUSTER TO WHICH THE FIRST ELIMINANT CURRENTLY BELONG 670
C
C           ITTT=FGRP(1)                                671
C           IT=LFFT(ITTT)                            672
C           LFFT(NONCP(LE))=IT                      673
C
C           THE IT-TH CLUSTER IS EXPANDED TO INCLUDE ELIMINANTS 674
C
C           NINCL(IT)=NINCL(IT)+1                  675
C           INCLUS(IT,NINCL(IT))=NCNCP(LF)        676
C           IF(NE.EQ.1) GO TO 3171                677
5011  DO 3172 J=JA,NE                      678
C
C           IU IS THE ACTIVITY NUMBER OF THE NEXT ELIMINANT 679
C           IF IU IS IN CLUSTER K, THEN CLUSTER K IS POOLED INTO CLUSTER IT 680
C
C           TU=EGRP(J)                                681
C           K=LFFT(IU)                            682
C           IF(IT.EQ.K) GO TO 3172                683
3182  NCLUS=NCLUS-1                          684
C           IW=NCLINC(K)
C           DO 3183 IA=1,IW                      685
C           LFFT(ICRITP(CLINCL(K,IA)))=IT        686
C           NCLINC(IT)=NCLINC(IT)+1            687
C
3183  CLINCL(IT,NCLINC(IT))=CLINCL(K,IA)    688
C           NCLINC(K)=0                            689
C           IW=NINCL(K)                          690
C           NINCL(K)=0                            691
C           DO 3184 IA=1,IW                      692
C           LFFT(INCLUS(K,IA))=IT                693
C           NINCL(IT)=NINCL(IT)+1            694
3184  INCLUS(IT,NINCL(IT))=INCLUS(K,IA)    695
C
3172  CONTINUE                                696
C
3171  CONTINUE                                697
C           IF(LF.LT.NNNCP) GO TO 3126          698
C
C           END OF POOLING BASED ON ELIMINANTS EXCEPT FOR THE FOLLOWING 699
C           DESCRIPTION                           700
C
C           WRITE(6,3173) NCLUS                  701
3173  FORMAT(1H1,05X,'THERE ARE ',I3,' CLUSTERS.') 702
C           DO 3176 I=1,KCP9                  703
C           IF(NINCL(I).EQ.0) GO TO 3176          704
C           TDD=NINCL(I)                        705
C
C           WRITE(6,3174) NINCL(I),I,(INCLUS(I,J),J=1,100) 706
C
3174  FORMAT(1H0,10X,'THERE ARE ',I3,' ACTIVITIES IN THE ',I3,'-TH CLUST 707
C

```

```

*ER. THEY ARE AS FOLLOWS:'./.20X.50(I3,'')
IDUCK=NCLINC(I)
WRITF(6,3175) NCLINC(I),(CLINCL(I,J),J=1,1DUCK)
3175 FORMAT(1H0,15X,I3,' CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER
*. THEY WERE AS FOLLOWS:'./.20X.50(I3,'')
3176 CONTINUE
IF (SAMSIZ.LE.0) WRITE(6,6045)
IF (SAMSIZ.GT.0) WRITE(6,6056) SAMSIZ,SAMSIZ
6055 FORMAT(5x,' THE FOLLOWING TABLES WERE DETERMINED CONSIDERING AT M
*0ST ',I8,' ACTIVITY CONFIGURATIONS PER CLUSTER.',/,11X,'IF THERE A
*RE NO MORE THAN ',I8,' ACTIVITY CONFIGURATIONS IN A CLUSTER.',/,11
*X,'THEN ALL ACTIVITY CONFIGURATIONS ARE EXPLICITLY CONSIDERED, AND
* NO SAMPLING IS DONE.')
6045 FORMAT (5X,' THE FOLLOWING TABLES WERE DETERMINED CONSIDERING ALL
* ACTIVITY CONFIGURATIONS.')
C
C      WARNING: THE SYSTEM SUBROUTINE CLOCK MAY NOT BE A PART OF
C      ALL SYSTEMS. XRAN NEEDS TO BE A RANDOM SEED.
C
CALL CLOCK(XRAN)
IYUTS = XRAN
WRITF (6,3238) IYUTS
3238 FORMAT (1H0,5X,'THE INITIALIZATION PARAMETER FOR ANY SAMPLING IS I
*Y = ',I10)
C
C      STATEMENT NUMBER 3124 MARKS THE END OF POOLING CLUSTERS BASED
C      ON LEFTOVERS AND ELIMINANTS
C
3124 CONTINUE
IF(COPTON1 .EQ.1) GO TO 999
C
C      THE FINAL CLUSTERS HAVE NOW BEEN DETERMINED
C      THE 2**NINCL(I) RUNS ARE NOW MADE FOR ALL I WITH NINCL(I)>0.
C
C
6008 IF(ICRCP.EQ.1) ICRCP=2
INDEXL=2
IF(COPTON2 .EQ.1) GO TO 87701
IF(ICRCP.EQ.2) GO TO 87701
DO 17701 I=1,M
17701 COT(I)=FL0(I)
87701 CONTINUE
IR=C
3200 IR=IR+1
IF(IP.GT.KCPB) GO TO 3208
IF(NINCL(IR).EQ.0) GO TO 3200
DO 6031 I=1,10
6031 MOMENT(IR,I) = 0.00
DO 9101 I=1,IEOF
9101 NLFFD(IR,I)=0.00
3310 IP = C
NIB(IP)=C
C
C      NIB = NUMBER OF ACTIVITY CONFIGURATIONS IN THE SAMPLE
C      NSAVE = VECTOR CONTAINING THE UPPER BOUNDS ON THE NETWORK
C              COMPLETION TIME DISTRIBUTION TO BE AVERAGED WITH THE
C              LOWER BOUNDS ON THE NETWORK TO YIELD THE AVERAGE NETWORK
C              COMPLETION TIME DISTRIBUTION.
C

```

```

C
C   IC=NINCL(IR)
C   IR=2**NINCL(IR)
C   IDOALL = 1 MEANS ALL ACTIVITY CONFIGURATIONS ARE EXPLICITLY
C           CONSIDERED.                                         780
C   IDCALL = 0 MEANS TO SAMPLE.                                781
C
C
C   IDOALL = 0
C   IF (SAMSIZ.LF.0.OR.SAMSIZ.GE.IB) IDOALL=1
C   DO 3222 I=1,M                                         782
3222 CTIME(I) = COT(I)
GO TO 20000                                              783
C
C   STATEMENT 3204 IS THE RE-ENTRY POINT FROM THE SIMPLEX
C   ALGORITHM WHEN CLUSTFR BASED BOUNDS ARE BEING COMPUTED
C
3204 CONTINUE                                              784
DO 6032 I=1,10                                         785
6032 MOMENT(IR,I) = MOMENT(IR,I) +(BIINV(1,NMMP1)**I)*SPROR
9900 X= BIINV(1,NMMP1)                                     786
X=X-1.00-10
I=0
6420 I=I+1
IF(X.GT.FD(I)) GO TO 6420
NLEFD(IR,I)=NLEFD(IR,I)+SPRUB
10001 CONTINUE                                             787
20000 IP=IP+1
NIB(IP)=NIB(IR)+1
C
C   GENERATE NEXT ACTIVITY CONFIGURATION TO BE EXPLICITLY
C   CONSIDERED.                                         788
C
IF (IDOALL.EQ.0) GO TO 20500
IF (IP.GT.IB) GO TO 3207
RANSAM = IP
GO TO 20501
20500 IF(NIR(IP).GT.SAMSIZ) GO TO 3207
IYUTS = IYUTS*65539
IF (IYUTS) 6210,6211,6211
6210 IYUTS = IYUTS+2147483647+1
6211 XRAM = IYUTS
XRAM = XRAM*.4656613F-9
RANSAM = XRAM*DFLOAT(IB-1) + 1
20501 CONTINUE                                              789
C
C   CONVERT THE RANDOM NUMBER, RANSAM, TO A BINARY NUMBER TO DEFINE AN
C   ACTIVITY CONFIGURATION.                               790
C
KRAM = RANSAM
SPROR=1.00
DO 8505 I=1,IC                                         791
IHALF = KRAM/2
IZ = KRAM - IHALF*2
L = INCLUS(IR,I)
CTIME(L) = IZ*FHT(L)-IZ*FLO(L) + FLO(L)
SPROR=SPROR*(IZ*PQ(L)+(1-IZ)*PP(L))*2.00
C
C   SPROR= 2**NINCL(IR) * THE PROBABILITY OF THIS CONFIGURATION
C

```

```

8505 KTRAN = 1HALF          840
      CALL BINVA(62800)        841
3207 NIR(IR)=NIB(IR)-1     842
      DO 6030 I=1,10          843
6030 MOMENT(IR,I) = MOMENT(IR,I)/NIB(IR) 844
      GO TO 3200             845
3208 WRITE (6,6362)          846
6362 FORMAT (1H1)            847
      IF (ICRCP,FQ.2) GO TO 6611 848
      DO 10000 J=1,10          849
      ITMAX=0                 850
      TMAX=0.                  851
      DO 5021 I=1,KCPB         852
      IF(NINCL(I).EQ.0) GO TO 5021 853
      IF (MOMENT(I,J).LT.TMAX) GO TO 5021 854
      TMAX = MOMENT(I,J)        855
      ITMAX=I                  856
5021 CONTINUE                857
      WRITE (6,5023) J,J,MOMENT(ITMAX,J) 858
5023 FORMAT (1H0,5X,'A LOWER BOUND, T-(',I2,';THFTA,LAMBDA), ON THE '
*,           I2 ,'-TH MOMENT OF THE NETWORK COMPLETION TIME = ',E15.5) 859
10000 CONTINUE                860
C                                     861
C                                     862
C                                     863
C                                     BEGIN THE PROCEDURE FOR DETERMINING UPPER BOUNDS ON THE 864
C                                     NETWORK COMPLETION TIME DISTRIBUTION 865
C                                     866
9011 CONTINUE                867
      DO 9007 IR=1,KCPB        868
      IF(NINCL(IR).EQ.0) GO TO 9007 869
      DO 9990 I=2,IEDF          870
      II=I-1                  871
9990 NLFFD(IR,I)=NLEFD(IR,I)+NLEFD(IP,II) 872
      WNNIR=DFLCAT(NIR(IR))    873
      DO 39990 I=1,IEDF          874
39990 NLFFD(IR,I)=NLEFD(IP,I)/WNNIR 875
9007 CONTINUE                876
      DO 10111 IR=1,KCPB        877
      IF(NINCL(IR).GT.0) GO TO 10112 878
10111 CONTINUE                879
10112 IRR=IP                 880
C                                     IRR = NON-EMPTY CLUSTER WITH THE SMALLEST INDEX 881
C                                     882
      DO 10119 IR=1,KCPB        883
      IF(NINCL(IR).EQ.0) GO TO 10119 884
      DO 10117 I=1,IEDF          885
      IF(NLEFD(IR,I).LT.NLEFD(IRR,I)) NLEFD(IRR,I)=NLFFD(IR,I) 886
10110 CCNTINUE                887
10119 CONTINUE                888
      WRITE (6,6264)             889
      WRITE(6,9423)              890
9423 FORMAT(1H0,5X,'AN UPPER BOUND ON THE NETWORK COMPLETION TIME DISTR 891
*IRTRITION: F+(.;THFTA;LAMHDA)") 892
      *IRTRITION: F+(.;THFTA;LAMHDA)") 893
      DO 9421 I=1,IEDF          894
      NSAVE(I)=NLEFD(IRR,I)      895
      X=NLEFD(IRR,I)            896
9421 WRITE(6,9422) FD(I),THFTA,LAMBDA,X 897
9422 FORMAT(17X,'F+(.,F15.5,'';',E15.5,';',F15.5,'') = ',E15.5) 898
C                                     899

```

C BEGIN THE PROCEDURE FOR DETERMINING UPPER BOUNDS ON THE MOMENTS 900
 C OF THE NETWORK COMPLETION TIME AND LOWER BOUNDS ON THE 901
 C DISTRIBUTION OF THE COMPLETION TIMES. 902
 C
 C THE UPPER BOUNDS, T+(R,THETA,LAMBDA), ARE NOW DETERMINED. 903
 C
 C
 C FOR THE SAKE OF NUMERICAL ACCURACY THE PERT PROBLEM 904
 C WITH NEW ACTIVITY TIMES IS INITIALLY SOLVED FROM SCRATCH 905
 C INSTEAD OF UPDATING AN OLD SOLUTION. AFTER THIS 906
 C REINITIALIZATION, THE REMAINING CRITICAL PATH TIMES ARE 907
 C DETERMINED BY UPDATING THIS SOLUTION. 908
 C
 C
 ICRCP=1 910
 DO 6001 I=1,M 911
 CTIME(I)=FHI(I) 912
 6001 COT(I)=FHI(I) 913
 GO TO 6010 914
 6011 WRITE (6,6264) 915
 6264 FORMAT(///) 916
 DO 10009 J=1,10 917
 ITMIN=IRR 918
 TMIN=MOMENT(IRR,J) 919
 DO 7101 I=1,KCPB 920
 IF(NINCL(I).EQ.0) GO TO 7101 921
 IF(MOMENT(I,J).GT.TMIN) GO TO 7101 922
 ITMIN=I 923
 TMIN=MOMENT(I,J) 924
 7101 CONTINUE 925
 10009 WRITE (6,6012) J,J,MOMENT(ITMIN,J) 926
 6012 FORMAT (1HO,5X,'AN UPPER BOUND, T+(.,I2,';THETA,LAMBDA), ON THE ' 927
 * ,I2,'-TH MOMENT OF THE NETWORK COMPLETION TIME = ',E15.5) 928
 DO 66900 IR=1,KCPB 929
 IF(NINCL(IR).EQ.0) GO TO 66900 930
 DO 6900 I=2,IEDF 931
 II=I-1 932
 6900 NLFFD(IR,I)=NLFFD(IR,I)+NLEFD(IR,II) 933
 WNNIB=DFLOAT(NIB(IR)) 934
 DO 49930 I=1,IEDF 935
 49990 NLFFD(IR,I)=NLEFD(IR,I)/WNNIB 936
 66900 CONTINUE 937
 DO 20115 IR=1,KCPB 938
 IF(NINCL(IR).EQ.0) GO TO 20115 939
 DO 20110 I=1,IEDF 940
 IF(NLFFD(IR,I).GT.NLFFD(IRR,I)) NLEFD(IRR,I)=NLFFD(IR,I) 941
 20110 CONTINUE 942
 20115 CONTINUE 943
 WRITE (6,6362) 944
 WRITE(6,6423)
 6423 FORMAT(1HO,5X, 'A LOWER BOUND ON THE NETWORK COMPLETION TIME DISTR 945
 * IBUTION: F-(.;THETA;LAMBDAA')) 946
 DO 6421 I=1,IEDF 947
 X=NLFFD(IRR,I) 948
 6421 WRITE(6,6422) FD(I),THETA,LAMBDA,X 949
 6422 FORMAT(17X,'F-(.,F15.5,.;',E15.5,';',E15.5,') = ',E15.5) 950
 C

```

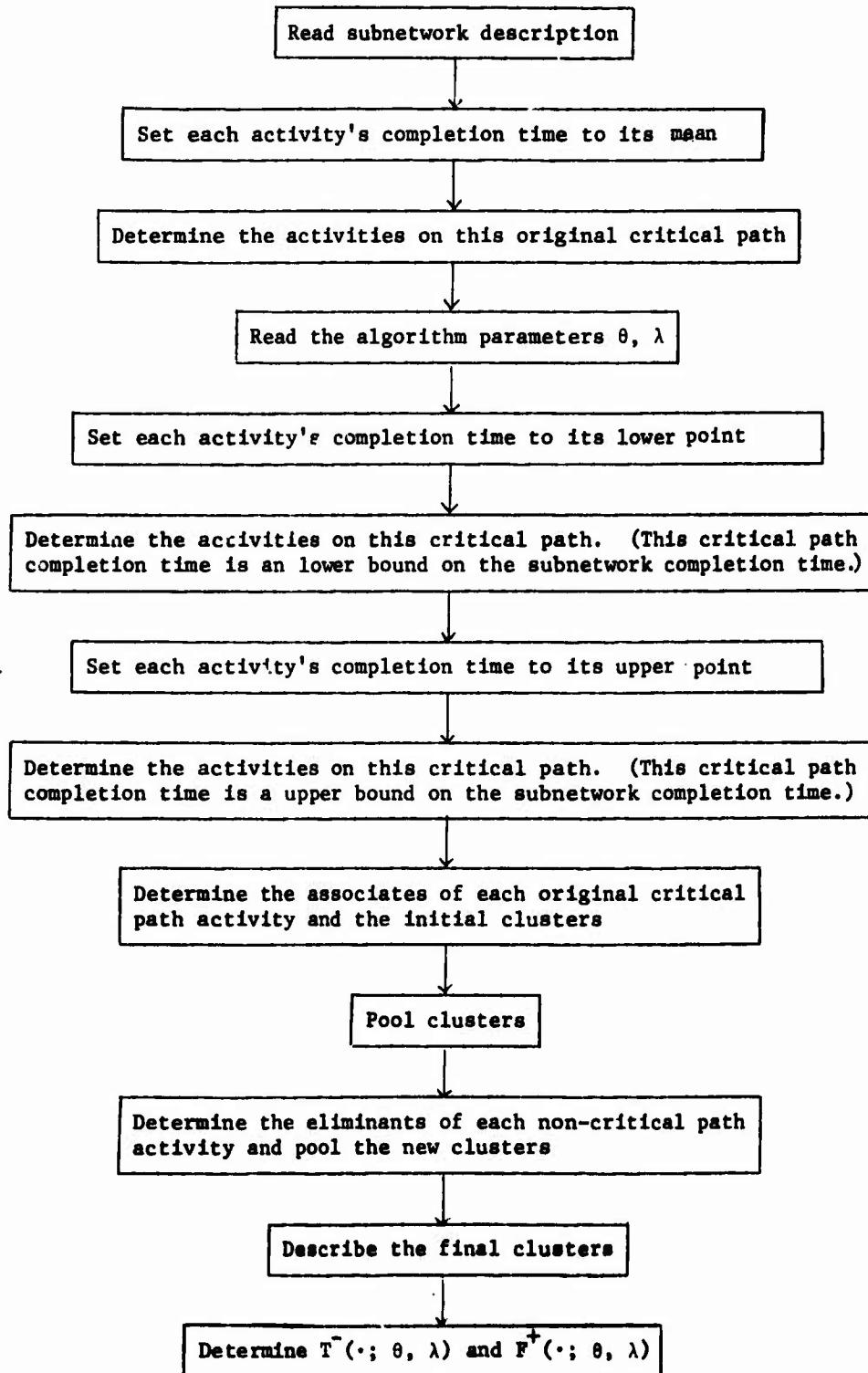
C      THE APPROXIMATE NETWORK COMPLETION TIME DISTRIBUTION      960
C
C      WRITE(6,6362)
C      WRITE(6,9472)
9472  FORMAT(1H0,5X,'AN APPROXIMATE NETWORK COMPLETION TIME DISTRIBUTION 961
*:'//,15X,'F(.,;THETA,LAMBDA) = .5 * ( F+(.,;THETA,LAMBDA) + F-(.,;TH 962
*ETA,LAMBDA) )',//) 963
      DO 9471 I=1,1EDF 964
      X=.5D0*NSAVE(I)      +.5D0*NLEFD(IRR,I) 965
9471  WRITE(6,9473) FD(I),THETA,LAMBDA,X 966
9473  FORMAT(17X,' F(.,E15.5,';',E15.5,';',E15.5,') = ',F15.5) 967
999   WRITE(6,850)
      STOP
      END
      SUBROUTINE BINVA(*)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON B1INV,RFDCOS,CTIME,XB1,INBASF,IHEAD,ITAIL,NMMP1,NMM,N,ISTAT
      COMMON M,MPI
      DIMENSION ISTAT(100),IHEAD(60),ITAIL(60),XB1(41)
      DIMENSION B1INV(41,41),INBASE(40),CTIME(100),RFDCOS(100) 971
C
C      UPDATE THE FIRST ROW OF B1INV AFTER CHANGING CTIME          972
C
      DO 1 I=2,NMMP1 973
      B1INV(1,I) = 0.0 974
      DO 1 J=2,NMMP1 975
      1 B1INV(1,I) = B1INV(1,I) + B1INV(J,I)*CTIME(INBASE(J-1)) 976
C
C      UPDATE VALUE OF THE OBJECTIVE FUNCTION                      977
C
      XB1(1) = B1INV(1,NMMP1) 978
      RETURN1
      END
      SUBROUTINE BINV1 (*,TMNFW,TMOLD,1D)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON B1INV,RFDCOS,CTIME,XB1,INBASE,IHEAD,ITAIL,NMMP1,NMM,N,ISTAT
      COMMON M,MPI
      DIMENSION ISTAT(100),IHEAD(60),ITAIL(60),XB1(41)
      DIMENSION B1INV(41,41),INBASE(40),CTIME(100),RFDCOS(100) 995
C
C      COMPUTE THE RFDCOS CORRESPONDING TO ONE CHANGE IN CTIME    996
C
      DO 2 I=1,NMM 1000
      2 IF(INBASE(I).EQ.1D) I=I+1 1001
      DIFF = TMNEW-TMOLD 1002
      TMNEW IS THE NEW TIME AND TMOLD IS THE OLD TIME CORRESPONDING 1003
      TO THE SINGLE CHANGE IN CTIME 1004
C
      DO 1 K=1,M 1005
      IF(ISTAT(K).EQ.1) GO TO 1 1006
      RFDCOS(K) = REDCOS(K)-DIFF*(B1INV(1,IHEAD(K)+1) - 1007
      * B1INV(1,ITAIL(K)+1)) 1008
      1 CONTINUE 1009
      DO 3 K=MPI,N 1010
      IF(ISTAT(K).EQ.1) GO TO 3 1011
      RFDCOS(K) = REDCOS(K) - DIFF*B1INV(1,K-M+1) 1012
      3 CONTINUE 1013
C
C      UPDATE THE FIRST ROW OF B1INV AFTER CHANGING CTIME          1014
C

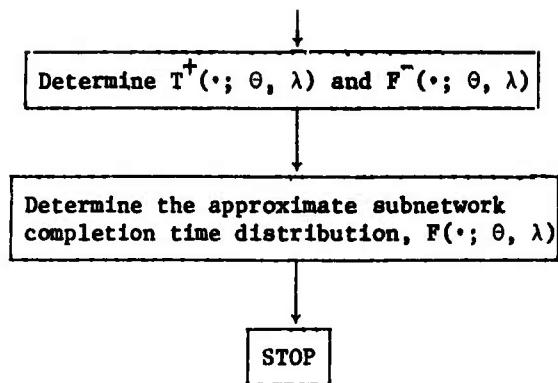
```

```
DO 10 I=2,NMMP1  
B1INV(1,I)=B1INV(1,I)+DIFF*B1INV(II,I)  
10 CONTINUE  
  
C C UPDATE VALUE OF THE OBJECTIVE FUNCTION  
C  
XB1(1) = B1INV(1,NMMP1)  
RETURN1  
END
```

1020
1021
1022
1023
1024
1025
1026
1027
1028

New Subnetwork Analysis Program: Flowchart





APPENDIX C

Original Subnetwork Analysis Program

The Original Subnetwork Analysis Program is an implementation and extension of the analytical procedure described in Section 3 of Technical Report No. 48. The basic required input is

- (a) an acyclic network with one source and one sink,
- (b) two points from each component activity's completion time distribution, and
- (c) specified values for the algorithm parameters θ and λ .

The output is mainly

- (a) upper and lower bounds on the moments of the network completion time, $T_r^+(\theta, \lambda)$ and $T_r^-(\theta, \lambda)$ $r = 1, 2, \dots, 10$;
- (b) upper and lower bounds on the distribution function of the network completion time, $F^+(\cdot; \theta, \lambda)$ and $F^-(\cdot; \theta, \lambda)$; and
- (c) an approximate network completion time distribution,
$$F(\cdot; \theta, \lambda) = 1/2[F^+(\cdot; \theta, \lambda) + F^-(\cdot; \theta, \lambda)].$$

The main extension of this program is the inclusion of an option to consider only a random sample of the 2^{n_c} activity time configurations for a cluster C instead of explicitly evaluating the critical path time for all of the 2^{n_c} activity time configurations.

The basic computational technique for determining critical path times is the Simplex Algorithm. This algorithm is applied to the dual problem. The Simplex Algorithm is used instead of the standard network analysis techniques because the Simplex Algorithm is ideally suited for the type of parametric programming required to evaluate several critical path times when only the activity times vary from one problem to the next.

A listing of the Original Subnetwork Analysis Program and a program flowchart are given at the end of this appendix.

Specific Input Instructions:

Card 1. Col. 1-3 : The number of activities in the network, Format (I3).

Col. 4-6 : The number of nodes in the network, Format (I3).

For each activity one card with:

Col. 11-15: The origin node of the activity, Format (I5).

Col. 21-25: The terminal node of the activity, Format (I5).

Col. 31-40: The lower point on the activity's completion time distribution, Format (F10.0)

Col. 41-50: The upper point on the activity's completion time distribution, Format (F10.0)

Next Card. Col. 1: OPTON1. OPTON1=1 implies that the program will terminate after the clusters have been formed on the basis of associates and eliminants. OPTON#1 implies that the program will follow the normal procedure.

Next Card. Col. 1-3: IEDF. The program computes an absolute upper and lower bound for the network completion time. This range is subdivided into IEDF equal parts and the approximate distribution function (F^+ , F^- , f) values are printed at each of these dividing points. IEDF would usually be between 10 and 100. IEDF, Format (I3).

Next Card. Col. 1-5 : θ, Format (F5.2).

Col. 6-10: λ, Format (F5.2)

Next Card. Col. 1-10: SAMSIZ. The number of activity time configurations to be randomly selected for explicit consideration in each cluster analysis. If $SAMSIZ < 0$ or $SAMSIZ > 2^{n_c}$, all activity time configurations will be explicitly considered - no random sampling will be done. Format (I10).

The nodes should be numbered 1, 2, ..., n with the source being number 1, the sink being number n, and the other node numbers being arbitrary. The activities should be numbered 1, 2, ... in any order desired.

Current Dimension Restrictions:

Currently the program is dimensioned for a maximum of

60 Activities

40 Nodes

25 Clusters

25 Activities/Cluster and IEDF \leq 500.

Example:

The program's input and output are illustrated in terms of the network in Figure C-1.

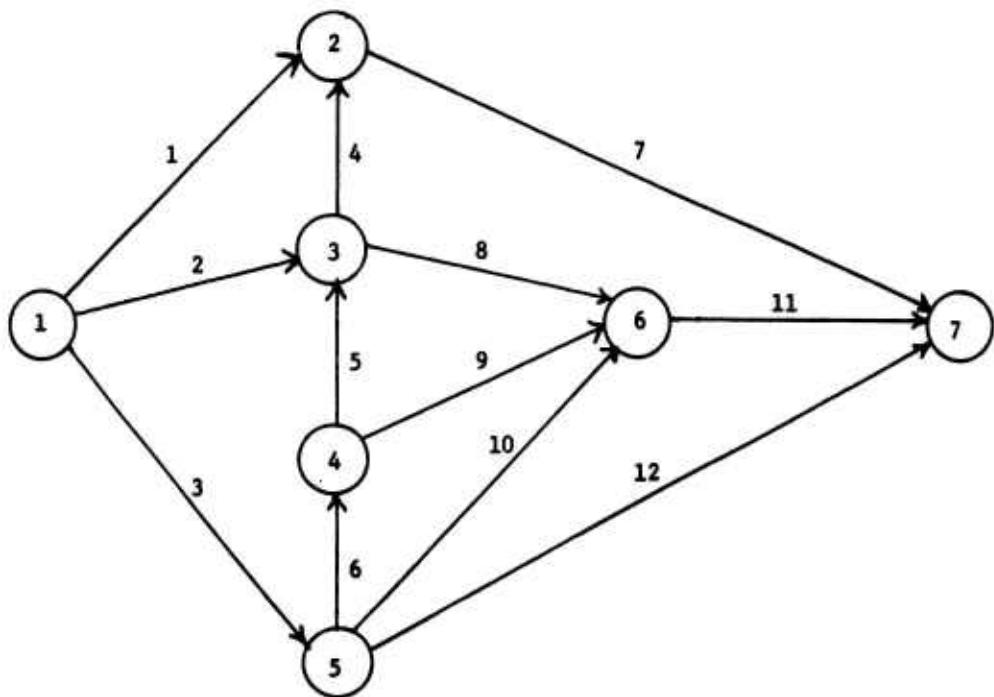


Figure C-1. Original Subnetwork Analysis Program Example Network

SAMPLE INPUT

69

012007

1	2	17.26	19.44
1	3	19.26	21.44
1	5	12.76	15.91
3	2	3.51	4.01
4	3	3.01	5.43
5	4	3.52	4.25
2	7	13.75	14.48
3	6	5.05	8.43
4	6	5.36	6.51
5	6	8.78	11.44
6	7	15.76	17.21
5	7	14.32	18.35

0

020

1. 1.

-1

SAMPLE OUTPUT

INITIAL INPUT

ACTIVITY	ORIGIN	TERMINAL	LOWER DFRC.	UPPER PERC.	AVERAGE	PERCENTILE DIFFERENCE
1	1	2	17.2600	19.4400	18.3500	2.1800
2	1	3	19.2600	21.4400	20.3500	2.1800
3	1	5	12.7600	15.9100	14.3350	3.1500
4	3	2	3.5100	4.7100	3.7600	0.5000
5	4	3	3.4100	5.4300	4.2250	2.4200
6	5	4	3.5200	4.2500	3.9850	0.7300
7	2	7	13.7500	14.4800	14.1150	0.7300
8	3	6	5.0500	8.4300	6.7400	3.3800
9	4	6	5.3600	6.5100	5.9350	1.1500
10	5	6	8.7800	11.4400	10.1100	2.6600
11	6	7	15.7600	17.2100	16.4650	1.4500
12	5	7	14.3200	18.3500	16.3350	4.0300

THE CRITICAL PATH TIME WHEN EACH ACTIVITY'S COMPLETION TIME IS SET EQUAL TO ITS AVERAGE IS = C.4566650 C2

THE 6 NODES ON THE CRITICAL PATH ARE AS FOLLOWS BEGINNING WITH THE TERMINAL NODE:
7. 6. 3. 4. 5. 1.

THE 5 CRITICAL ACTIVITIES ARE AS FOLLOWS BEGINNING WITH THE TERMINAL ACTIVITY:

11. 9. 5. 6. 3.

THETA = 0.102000 C1 LAMRCA = C.102000 C1

A LOWER BOUND ON THE EXPFCED CRITICAL PATH TIME IS = C.401000 C2

THE 4 NODES ON THE CRITICAL PATH ARE AS FOLLOWS BEGINNING WITH THE TERMINAL NODE:
7. 6. 3. 4. 5. 1.

THE 5 CRITICAL ACTIVITIES ARE AS FOLLOWS BEGINNING WITH THE TERMINAL ACTIVITY:

11. 9. 5. 6. 3.

A UPPER BOUND ON THE EXPFCED CRITICAL PATH TIME IS = C.512300 C2

THE 6 NODES ON THE CRITICAL PATH ARE AS FOLLOWS BEGINNING WITH THE TERMINAL NODE:
7. 6. 3. 4. 5. 1.

THE 5 CRITICAL ACTIVITIES ARE AS FOLLOWS BEGINNING WITH THE TERMINAL ACTIVITY:

11. 9. 5. 6. 3.

THE ASSOCIATES ARE NOW IDENTIFIED:

THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE 1-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY 11. IS = C
THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE 2-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY 8. IS = C
THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE 3-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY 5. IS = 1
THE ACTIVITIES IN THE ASSOCIATE GROUP ARF AS FOLLOWS
2.

THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE 4-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY 6. IS = 0
THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE 5-TH CRITICAL PATH ACTIVITY. I.E. ACTIVITY 3. IS = 1
THE ACTIVITIES IN THE ASSOCIATE GROUP ARF AS FOLLOWS

2.

THERE ARE 4 NINEMONY CLUSTERS AFTER POOLING ON THE BASIS OF ASSOCIATES ONLY.

THE ACTIVITIES IN THE 1-TH CLUSTER ARE AS FOLLOWS:

11.

THE ACTIVITIES IN THE 2-TH CLUSTER ARE AS FOLLOWS:

3.

THE ACTIVITIES IN THE 3-TH CLUSTER ARE AS FOLLOWS:

5, 2, 3.

THE ACTIVITIES IN THE 4-TH CLUSTER ARE AS FOLLOWS:

6.

THE CLUSTER IN WHICH EACH ACTIVITY BELONGS:

(ZERO IMPLIES THAT THE ACTIVITY IS NOT IN ANY CLUSTER)

THE 1-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 2-TH ACTIVITY IS IN THE 3-TH CLUSTER

THE 3-TH ACTIVITY IS IN THE 3-TH CLUSTER

THE 4-TH ACTIVITY IS IN THE 5-TH CLUSTER

THE 5-TH ACTIVITY IS IN THE 3-TH CLUSTER

THE 6-TH ACTIVITY IS IN THE 4-TH CLUSTER

THE 7-TH ACTIVITY IS IN THE 0-TH CLUSTER

THE 8-TH ACTIVITY IS IN THE 2-TH CLUSTER

THE 9-TH ACTIVITY IS IN THE 3-TH CLUSTER

THE 10-TH ACTIVITY IS IN THE 4-TH CLUSTER

THE 11-TH ACTIVITY IS IN THE 1-TH CLUSTER

THE 12-TH ACTIVITY IS IN THE 3-TH CLUSTER

THERE ARE 6 ACTIVITIES NOT IN ANY CLUSTER YET.

TMF ELIMINANTS OF EACH NON-CRITICAL-PATH ACTIVITY AND NOW DETERMINED:
 TMF: A/F 7 ACTIVITIES NOT IN THE CRITICAL PATH. THEY ARE AS FOLLOWS:

1.	1
2.	2
3.	4
4.	7
5.	9
6.	10
7.	12

TMF COMPLETION TIME FOR THE 1-TH ACTIVITY HAS BEEN CHANGED TO C.0.225300 02
 THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 1

TMF COMPLETION TIME FOR THE 2-TH ACTIVITY HAS BEEN CHANGED TO C.0.225300 02
 THERE ARE 3 ELIMINANTS CORRESPONDING TO ACTIVITY 2
 THE 1-TH ELIMINANT CORRESPONDING TO ACTIVITY 2 IS ACTIVITY 5.
 THE 2-TH ELIMINANT CORRESPONDING TO ACTIVITY 2 IS ACTIVITY 6.
 THE 3-TH ELIMINANT CORRESPONDING TO ACTIVITY 2 IS ACTIVITY 3

TMF COMPLETION TIME FOR THE 4-TH ACTIVITY HAS BEEN CHANGED TO C.0.425900 01
 THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 4

TMF COMPLETION TIME FOR THE 7-TH ACTIVITY HAS BEEN CHANGED TO C.0.149450 02
 THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 7

TMF COMPLETION TIME FOR THE 9-TH ACTIVITY HAS BEEN CHANGED TO C.0.768500 01
 THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 9

TMF COMPLETION TIME FOR THE 10-TH ACTIVITY HAS BEEN CHANGED TO C.0.127700 02
 THERE ARE 0 ELIMINANTS CORRESPONDING TO ACTIVITY 10

TMF COMPLETION TIME FOR THE 12-TH ACTIVITY HAS BEEN CHANGED TO C.0.203050 02

THERE ARE 6 ELIMINANTS CORRESPONDING TO ACTIVITY 12

THERE ARE 3 CLUSTERS. THERE ARE 3 ACTIVITIES IN THE 1-TH CLUSTER. THEY ARE AS FOLLOWS:

1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:

1.

THERE ARE 1 ACTIVITIES IN THE 2-TH CLUSTER. THEY ARE AS FOLLOWS:

a.

1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:

2.

THERE ARE 4 ACTIVITIES IN THE 3-TH CLUSTER. THEY ARE AS FOLLOWS:

5. 2. 3. 6.

1 CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER. THEY WERE AS FOLLOWS:

3. 5. 4.

THE NUMBER OF PERCENTILE COMBINATIONS EXPLICITLY CONSIDERED IN DETERMINING THE UPPER BOUNDS AND LOWER BOUNDS ON THE NETWORK COMPLETION TIME DISTRIBUTION AND THE UPPER BOUNDS ON ITS MOMENTS IS EQUAL TO -1.
THE INITIALIZATION PARAMETER FOR THE SAMPLING IS IV = 77

**COPY AVAILABLE TO DOC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION**

THE FOLLOWING TABLE WAS COMPUTED CONSIDERING ALL PERCENTILE COMBINATIONS.

A LCDFP SCUND. T-(1:THETA.LAMRDA).	ON THF	1-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.45868D 02
A LOMFR SCUND. T-(2:THFTA.LAMRDA).	ON THF	2-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.21099D 04
A LOMFR SCUND. T-(3:THFTA.LAMRDA).	ON THF	3-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.97064D 05
A LOMFR SCUND. T-(4:THETA.LAMRDA).	ON THF	4-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.44742D 07
A LOWER SCUND. T-(5:THETA.LAMRDA).	ON THF	5-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.20655D 09
A LOWER SCUND. T-(6:THETA.LAMRDA).	ON THF	6-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.95494D 10
A LOWER SCUND. T-(7:THETA.LAMRDA).	ON THF	7-TH MOMENT OF THF NETWORK COMPLFTION TIME =	0.44215D 12
A LOWER SCUND. T-(8:THETA.LAMRDA).	ON THF	8-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.20533D 14
A LCDFP SCUND. T-(9:THETA.LAMRDA).	ON THE	9-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.95211D 15
A LCDFP SCUND. T-(10:THFTA.LAMRDA).	ON THE	10-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.44279D 17

THE FOLLOWING TABLE WAS COMPUTED CONSIDERING ALL PERCENTILE COMBINATIONS.

AN UPPER BOUND. T+(1:THETA.LAMRDA).	ON THE	1-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.45868D 02
AN UPPER BOUND. T+(2:THETA.LAMRDA).	ON THE	2-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.21123D 04
AN UPPER BOUND. T+(3:THETA.LAMRDA).	ON THF	3-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.97530D 05
AN UPPER BOUND. T+(4:THETA.LAMRDA).	ON THE	4-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.4517CD 07
AN UPPER BOUND. T+(5:THETA.LAMRDA).	ON THE	5-TH MOMENT OF THE NETWORK COMPLFTION TIME =	0.20983D 09
AN UPPER BOUND. T+(6:THETA.LAMRDA).	ON THE	6-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.97769D 10
AN UPPER BOUND. T+(7:THFTA.LAMRDA).	ON THE	7-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.4568AD 12
AN UPPER BOUND. T+(8:THFTA.LAMRDA).	ON THE	8-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.21412D 14
AN UPPER BOUND. T+(9:THFTA.LAMRDA).	ON THE	9-TH MOMENT OF THE NETWORK COMPLETION TIME =	0.10064D 16
AN UPPER BOUND. T+(10:THFTA.LAMRDA).	ON THE	10-TH MOMENT OF THF NETWORK COMPLETION TIME =	0.47434D 17

THE FOLLOWING TABLE WAS COMPUTED CONSIDERING ALL PERCENTILE COMBINATIONS.

A LOWER BOUND ON THE NETWORK COMPLETION TIME DISTRIBUTION: F-(\cdot :THETA:LAMBDA)

F-(\cdot)	0.4C657D 02:	0.10000 01:	0.10000 C1:	0.10000 C1 = 0.15625D-01
F-(\cdot)	0.41213D 02:	0.10000 01:	0.10000 C1:	0.10000 C1 = 0.3125C0-01
F-(\cdot)	0.4177CD 02:	0.10000 01:	0.10000 C1:	0.46875D-01
F-(\cdot)	0.42326D 02:	0.10000 01:	0.10000 C1:	0.375CD-01
F-(\cdot)	0.42983D 02:	0.10000 01:	0.10000 C1:	0.1250CD 00
F-(\cdot)	0.43439D 02:	0.10000 01:	0.10000 C1:	0.1875C0 00
F-(\cdot)	0.43996D 02:	0.10000 01:	0.10000 C1:	0.2968RD 00
F-(\cdot)	0.44552D 02:	0.10000 01:	0.10000 C1:	0.3125C0 00
F-(\cdot)	0.451CD 02:	0.10000 01:	0.10000 C1:	0.39163D 00
F-(\cdot)	0.45665D 02:	0.10000 01:	0.10000 C1:	0.46875D 00
F-(\cdot)	0.46221D 02:	0.10000 01:	0.10000 C1:	0.53125D 00
F-(\cdot)	0.46778D 02:	0.10000 01:	0.10000 C1:	0.625C0D 00
F-(\cdot)	0.47334D 02:	0.10000 01:	0.10000 C1:	0.6475CD 00
F-(\cdot)	0.47871D 02:	0.10000 01:	0.10000 C1:	0.78125D 00
F-(\cdot)	0.48447D 02:	0.10000 01:	0.10000 C1:	0.84375D 00
F-(\cdot)	0.49004D 02:	0.10000 01:	0.10000 C1:	0.8750D 00
F-(\cdot)	0.4956CD 02:	0.10000 01:	0.10000 C1:	0.90625D 00
F-(\cdot)	0.5C117D 02:	0.10000 01:	0.10000 C1:	0.93750D 00
F-(\cdot)	0.50673D 02:	0.10000 01:	0.10000 C1:	0.96875D 00
F-(\cdot)	0.5123CD 02:	0.10000 01:	0.10000 C1:	0.10000 01

THE FOLLOWING TABLE WAS COMPUTED CONSIDERING ALL PERCENTILE COMBINATIONS.

AN UPPER BOUND ON THE NETWORK COMPLETION TIME DISTRIBUTION: F+(\cdot :THETA:LAMBDA)

F+(\cdot)	0.4C657D 02:	0.10000 01:	0.10000 C1:	0.10000 C1 = 0.15625D-01
F+(\cdot)	0.41213D 02:	0.10000 01:	0.10000 C1:	0.3125C0-01
F+(\cdot)	0.4177CD 02:	0.10000 01:	0.10000 C1:	0.46875D-01
F+(\cdot)	0.42326D 02:	0.10000 01:	0.10000 C1:	0.375CD-01
F+(\cdot)	0.42983D 02:	0.10000 01:	0.10000 C1:	0.1250CD 00
F+(\cdot)	0.43439D 02:	0.10000 01:	0.10000 C1:	0.1875C0 00
F+(\cdot)	0.43996D 02:	0.10000 01:	0.10000 C1:	0.2968RD 00
F+(\cdot)	0.44552D 02:	0.10000 01:	0.10000 C1:	0.3125C0 00
F+(\cdot)	0.451CD 02:	0.10000 01:	0.10000 C1:	0.39163D 00
F+(\cdot)	0.45665D 02:	0.10000 01:	0.10000 C1:	0.46875D 00
F+(\cdot)	0.46221D 02:	0.10000 01:	0.10000 C1:	0.53125D 00
F+(\cdot)	0.46778D 02:	0.10000 01:	0.10000 C1:	0.625C0D 00
F+(\cdot)	0.47334D 02:	0.10000 01:	0.10000 C1:	0.6475CD 00
F+(\cdot)	0.47871D 02:	0.10000 01:	0.10000 C1:	0.79125D 00
F+(\cdot)	0.48447D 02:	0.10000 01:	0.10000 C1:	0.84375D 00
F+(\cdot)	0.49004D 02:	0.10000 01:	0.10000 C1:	0.90625D 00
F+(\cdot)	0.4956CD 02:	0.10000 01:	0.10000 C1:	0.93750D 00
F+(\cdot)	0.50673D 02:	0.10000 01:	0.10000 C1:	0.96875D 00
F+(\cdot)	0.5123CD 02:	0.10000 01:	0.10000 C1:	0.10000 01

AN APPROXIMATE NETWORK COMPLETION TIME DISTRIBUTION:

$$F(+;\theta\text{ETA},\lambda\text{BDA}) = .5 * (F(+;\theta\text{ETA},\lambda\text{BDA}) + F(-;\theta\text{ETA},\lambda\text{BDA}))$$

F(0+406570	02:	0.10000D	01:	0.10000D	C1)	=	0.15625D-01
F(0+412130	02:	0.10000D	01:	0.10000D	C1)	=	0.3175CD-01
F(0+417700	02:	0.10000D	01:	0.10000D	C1)	=	0.46875D-01
F(0+42326D	02:	C.10000D	01:	C.10000D	C1)	=	C.9375CD-C1
F(0+42683D	02:	C.10000D	01:	C.10000D	C1)	=	0.12500D 00
F(0+43419D	02:	0.10000D	01:	0.10000D	C1)	=	0.1675CD 00
F(0+43996D	02:	0.10000D	01:	0.10000D	C1)	=	0.29688D 00
F(0+44552D	02:	0.10000D	01:	0.10000D	C1)	=	0.31250D 00
F(0+45109D	02:	C.10000D	01:	C.10000D	C1)	=	C.32063D 00
F(0+45665D	02:	C.10000D	01:	0.10000D	C1)	=	0.46875D 00
F(0+46221D	02:	C.10000D	01:	C.10000D	C1)	=	0.53125D 00
F(0+46778D	02:	C.10000D	01:	C.10000D	C1)	=	0.62500D 00
F(0+47334D	02:	0.10000D	01:	0.10000D	C1)	=	0.68750D 00
F(0+47891D	02:	C.10000D	01:	2.10399D	C1)	=	0.78125D 00
F(0+48447D	02:	C.10000D	01:	C.10000D	C1)	=	0.84375D 00
F(0+49004D	02:	0.10000D	01:	2.10000D	C1)	=	0.97500D 00
F(0+49560D	02:	0.10000D	01:	0.10000D	C1)	=	0.93625D 00
F(0+50117D	02:	0.10000D	01:	0.10000D	C1)	=	0.93750D 00
F(0+50673D	02:	C.10000D	01:	C.10000D	C1)	=	0.96875D 00
F(0+5123CD	02:	0.10000D	01:	2.10000D	C1)	=	0.12000D 01

```

C ORIGINAL SUBNETWORK ANALYSIS PROGRAM

C IMPLICIT REAL*8 (A-H,O-Z)
C FOR THE SAKE OF IDENTIFYING THE APPROPRIATE DIMENSIONS, LET
C M = THE NUMBER OF ACTIVITIES IN THE NETWORK
C NMM = NUMBER OF NODES IN THE NETWORK
C NMMP1 = NMM + 1
C N = M + NMM
C L = THE LENGTH OF THE CRITICAL PATH
C C = THE MAXIMUM NUMBER OF BRANCHES IN A CLUSTER
C IEDF = THE NUMBER OF DIVISIONS IN THE EMPIRICAL
C DISTRIBUTION FUNCTION

C
C INTEGER TAIL( M ),HEAD( M ),ASSGRP( L,L ),CLINCL(L,L),EGRP(L)
C DIMENSION NINCL( C ),INCLUS( L, C ),NCLINC( L )
C DIMENSION NLEFD( IEDF ),FD( IEDF ),NSAVE( IEDF )
C DIMENSION IZZ( C ),AVG( L ),THAT(L)
C DIMENSION INBASE(NMM),XNODE(NMM)
C DIMENSION LEFT(M ),LEFTO(M ),NONCP(M )
C DIMENSION XB1(NMMP1),Y1(NMMP1),REDCOS(N),ISTAT(N)
C DIMENSION ICRTIP(L),NINAG(M),ICRITN(L+1),CTIME(N),COT(N)
C DIMENSION KB(L),IBB(L),F25(M),F75(M),SIGMA(M),B1INV(NMMP1,NMMP1)
C REAL      MOMENT(L,10)

C OF COURSE THESE DIMENSIONS ARE MERELY UPPER BOUNDS

C
C COMMON B1INV,REDCOS,CTIME,XB1,INBASF,HEAD,TAIL,NMMP1,NMM,N,ISTAT
C COMMON M,MPI
C INTEGER TAIL(60),HEAD(60),ASSGRP(25,25),CLINCL(25,25),EGRP(25)
C INTEGER SAMSIZ,RANSAM
C DIMENSION NINCL(25),INCLUS(25,25),NCLINC(25)
C DIMENSION FD(500),NLEFD(500),NSAVE(500)
C DIMENSION AVG(25),THAT(25)
C DIMENSION INBASE(40)
C DIMENSION XNODE(40)
C DIMENSION XB1(41),Y1(41),REDCOS(100),ISTAT(100)
C DIMENSION B1INV(41,41),KB(25),IBB(25),FLO(60),FHI(60),SIGMA(60)
C DIMENSION ICRTIP(25),NINAG(50),ICRITN(26),COT(100),CTIME(100)
C DIMENSION LEFT(60),LEFTO(60),NONCP(60)
C REAL*8 LAMBDA,MOMENT(25,10)
C
C M = THE NUMBER OF ACTIVITIES IN THE NETWORK
C NMM = THE NUMBER OF NODES IN THE PERT NETWORK
C READ(5,100)  M,NMM
100 FORMAT(2I3)
C N=NMM+M
C MP1=M+1
C NMMP1=NMM+1

C THE ACTIVITIES ARE DESCRIBED IN TERMS OF THEIR NODES
C II=THE TAIL NODE, THE ORIGIN NODE
C JJ=THE HEAD NODE, THE TERMINAL NODE
C FLO = THE LOWER PERCENTILE
C FHI = THE UPPER PERCENTILE
C SIGMA = FLO - FHI

DO 610  I=1,M

```

```

READ(5,2501) II,JJ,FLO(I ),FHI(I )
SIGMA(I ) = FHI(I )-FLO(I )
2501 FORMAT(10X,1S,5X,1S, 5X,F10.0,F10.0)
COT(I ) = (FLO(I )+FHI(I ))/2.
CTIME(I ) = COT(I )
TAIL(I ) = II
610 HEAD(I )=JJ
C          COT(M)      = THE ORIGINAL RIGHT HAND SIDES, I.E. THE AVERAGE
C          OF FLO AND FHI
DO 55610 I=MP1,N
55610 CTIMF(I ) = 0.
C
C          OPTION1 =1 IMPLIES THAT THE PROGRAM WILL TERMINATE AFTER
C          THE CLUSTERS HAVE BEEN FORMED. NO BOUNDS ON
C          THE PROJECT COMPLETION TIME MOMENTS OR
C          DISTRIBUTION WILL BE DETERMINED.
C          OPTION1 NOT= 1 IMPLIES THAT THE NORMAL PROCEDURE WILL BE
C          FOLLOWED.
C
READ(5,77551) OPTION1
77551 FORMAT(10I1)
IF(OPTION1 .EQ.1) WRITE(6,77552)
77552 FORMAT(1H1,10X,'OPTION1=1 AND THE PROGRAM WILL TERMINATE AFTER THE
* CLUSTERS HAVE BEEN FORMED.',/,11X,'NO BOUNDS ON THE PROJECT COMPL
*ETITION TIME MOMENTS OR DISTRIBUTION WILL BE DETERMINED.')
READ(5,100) IEDF
WRITE(6,2700)
2700 FORMAT(1H1,15X,'INITIAL INPUT')
WRITE(6,2701)
2701 FORMAT(1H0,1CX,'ACTIVITY ORIGIN TERMINAL LOWER PERC. UPPER P
*ERC. AVERAGE PERCENTILE DIFFERENCE')
DO 2704 I=1,M
2704 WRITE(6,2702) I, TAIL(I ),HEAD(I ),FLO(I ),FHI(I ),CTIME(I ),SIGMA(I )
2702 FORMAT(1H ,13X,13.5X,13.7X,13.5X,F10.4,3X,F10.4,3X,F10.4,4X,F10.4)
C
C          THE FOLLOWING INDICATORS ARE USED:
C
C          IPARM = 1 IMPLIES THE CRITICAL PATH TIME WHEN ALL ACTIVITY
C          COMPLETION TIMES ARE SET EQUAL TO THEIR AVERAGES IS
C          BEING DETERMINED
C          IPARM = 2 IMPLIES THAT THE LOWER BOUND ON THE COMPLETION TIME
C          FOR THE SUBNETWORK IS BEING DETERMINED
C          IPARM = 3 IMPLIES THAT THE UPPER BOUND ON THE COMPLETION TIME
C          FOR THE SUBNETWORK IS BEING DETERMINED
C          IPARM > 3 WHEN INDEXL=0 IMPLIES THAT THE ASSOCIATES ARE
C          BEING DETERMINED
C
C          INDEXL=0 IMPLIES THAT INITIAL CLUSTERS ARE STILL BEING FORMED
C          INDEXL=1 IMPLIES THAT THE LEFTOVERS, THEIR ELIMINANTS, AND
C          POOLFD CLUSTERS ARE BEING DETERMINED
C          INDEXL=2 IMPLIES THAT THE 2**NINCL( ) RUNS FOR EACH CLUSTER
C          ARE BEING MADE AND AVERAGED
C
C          ICRCP = 0 IMPLIES THAT THE PROCEDURE FOR DETERMINING UPPER
C          BOUNDS ON THE MOMENTS OF THE NETWORK COMPLETION TIME
C          AND LOWER BOUNDS ON THE DISTRIBUTION OF COMPLETION
C          TIMES HAS NOT BEEN BEGUN

```

C ICBCP = 1 IMPLIES THAT THE PROCEDURE FOR DETERMINING UPPER 120
 C BOUNDS ON THE MOMENTS OF THE NETWORK COMPLETION TIME 121
 C AND LOWER BOUNDS ON THE DISTRIBUTION OF COMPLETION 122
 C TIMES IS BEING INITIALIZED 123
 C ICBCP = 2 IMPLIES THAT THE PROCEDURE FOR DETERMINING UPPER 124
 C BOUNDS ON THE MOMENTS OF THE NETWORK COMPLETION TIME 125
 C AND LOWER BOUNDS ON THE DISTRIBUTION OF COMPLETION 126
 C TIMES IS BEING CARRIED OUT 127
 C
 C IDLB = 0 IMPLIES THAT THE PROCEDURE FOR DETERMINING A UPPER 128
 C BOUND ON THE NETWORK COMPLETION TIME DISTRIBUTION HAS NOT 129
 C BEGUN 130
 C IDLB = 1 IMPLIES THAT THE UPPER BOUND ON THE NETWORK 131
 C COMPLETION TIME DISTRIBUTION IS BEING DETERMINED 132
 C
 C IPARM=1 133
 C INDFXL=0 134
 C ICBCP=0 135
 C IDLB = 0 136
 6010 CONTINUE 137
 DO 104 I=1,NMM 138
 104 INRASF(I)=M+I 139
 DO 2001 J=1,M 140
 2001 ISTAT(J)=0. 141
 DO 2002 J=MP1,N 142
 2002 ISTAT(J)=1 143
 DO 10 IT=1,NMMP1 144
 DO 12 L=1,NMMP1 145
 12 B1INV(L,II) = 0. 146
 10 B1INV(II,II) = 1. 147
 DO 30 I=1,NMM 148
 30 XR1(I) = 0. 149
 XB1(NMMP1) = 1. 150
 TOLR1=1.0D-10 151
 C
 C START THE SIMPLEX ALGORITHM 152
 C SOLVE THE DUAL PROBLEM 153
 C THE NUMBER OF VARIABLES IS M REAL + NMM SLACKS 154
 C FOR A TOTAL OF N VARIABLES 155
 C
 350 CONTINUE 156
 . 2800 DO 23 J=1,N 157
 RATS = 0. 158
 IF (ISTAT(J).EQ.1) GO TO 52800 159
 IF (J.GT.M) GO TO 22 160
 RATS = -B1INV(1,HEAD(J)+1)+B1INV(1,TAIL(J)+1) + CTIME(J) 161
 GO TO 52800 162
 22 RATS = -B1INV(1,J-M+1) 163
 52800 RFDCOS(J)= RATS 164
 23 CONTINUE 165
 22800 CONTINUE 166
 1RMAX=1 167
 RMAX=RFDCOS(1) 168
 DO 24 J=2,N 169
 IF(RFDCOS(J) .LT. RMAX) GO TO 24 170
 RMAX=RFDCOS(J) 171
 1RMAX=J 172
 CONTINUE 173
 24 IF(RMAX .LT. TOLR1) GO TO 401 174
 175
 176
 177
 178
 179

```

22824 CONTINUE 180
    DO 26 L=1,NMMP1 181
    IF (IRMAX.GT.M) GO TO 50026 182
    Y1(L) = -B1INV(L, TAIL(IRMAX)+1)+B1INV/L, HEAD(IRMAX)+1) 183
    GO TO 26 184
50026 Y1(L) = B1INV(L, IRMAX-M+1) 185
26 CONTINUE 186
    Y1(1) = Y1(1) - CTIME(IRMAX) 187
    NUMBER=0 188
    DO 27 L=2,NMMP1 189
27    IF(Y1(L).LE. TOLR1) NUMBER=NUMBER+1 190
    IF(NUMBER.EQ. NMM) GO TO 403 191
    RMIN=.99D 20 192
    IRMIN=0. 193
    DO 32 II=2,NMMP1 194
    IF(Y1(II).LE. TOLR1) GO TO 32 195
    RATS =XB1(II)/Y1(II) 196
    RR=RATS-RMIN 197
    IF(RR.GE. 0.00) GO TO 32 198
    RMIN=RATS 199
    IRMIN=II 200
32    CONTINUE 201
    DO 33 J=2,NMMP1 202
    WW=B1INV(IRMIN,J)/Y1(IRMIN) 203
    DO 37 L=1,NMMP1 204
37    B1INV(L,J)=B1INV(L,J)-WW*Y1(L) 205
33    B1INV(IRMIN,J)=WW 206
C 207
C      UPDATE THE BASIC VARIABLES: INBASE AND XB1 208
C 209
    ISTAT(INBASE(IRMIN-1))=0 210
    ISTAT(IRMAX)=1 211
    INBASE(IRMIN-1)=IRMAX 212
    W=XB1(IRMIN)/Y1(IRMIN) 213
    DO 38 I=1,NMMP1 214
38    XB1(I)=XB1(I)-Y1(I)*W 215
    XB1(IRMIN)=W 216
    GO TO 350 217
403    WRITE(6,530) 218
530    FORMAT(1H0,5X,'NO FEASIBLF SOLUTION EXISTS. CHECK YOUR INPUT DATA 219
*.*') 220
    WRITE(6,850) 221
850    FORMAT(1H1) 222
    GO TO 999 223
C 224
C      END OF THE SIMPLEX ALGORITHM 225
C 226
401    CONTINUE 227
C 228
C      KKK= THE NUMBER OF NODES ON THE CRITICAL PATH 229
C      KA(L)= THF L-TH NODE IN THE CRITICAL PATH, COUNTING BACKWARDS 230
C          FROM THE TERMINAL NODE 231
C      KKB= THE NUMBER OF ACTIVITIES ON THE CRITICAL PATH 232
C      IBA(L)= THE L-TH ACTIVITY ON THE CRITICAL PATH, COUNTING 233
C          BACKWARDS FROM THF TERMINAL NODE 234
C 235
    IF(ICBCP.EQ.1) GO TO 6008 236
    IF(INDFXL.EQ.2) GO TO 3204 237
C 238
C      INBASE IS A SFT OF M INTEGER VARIABLEFS WHICH INDICATE THE 239

```

**COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION**

C COMPOSITION OF THE CURRENT BASIS. FOR EXAMPLE. 240
C INBASE(K) = 7 IMPLIES THAT THE K-TH COLUMN IN THE BASIS B 241
C CORRESPONDS TO THE 7-TH VARIABLE 242
C
C
C ISTAT INDICATES THE BASIC STATUS OF EACH VARIABLE 243
C ISTAT(K) = 1 IMPLIES THAT THE K-TH VARIABLE IS IN THE 244
C DUAL BASIS 245
C ISTAT(K) = 0 IMPLIES THAT THE K-TH VARIABLE IS NOT IN THE 246
C DUAL BASIS 247
C
C
C THE FOLLOWING STATEMENTS DETERMINE THE NODES AND ACTIVITIES ON 248
C THE CRITICAL PATH 249
C
C
C THE DUAL SOLUTION IMPLIES THE FOLLOWING OPTIMAL SOLUTION TO THE 250
C PRIMAL PERT PROBLEM. HOWEVER SOME OF THE NODE TIMES(OTHER THAN 251
C THE LAST ONE) MAY BE HIGHER THAN NECESSARY. THUS IN 252
C DETERMINING THE CRITICAL PATH AN ALTERNATIVE OPTIMAL SOLUTION 253
C MAY HAVE TO BE IDENTIFIED. 254
C R1INV IS NOT CHANGED. 255
C
C DO P3002 I=1,NMM 256
83002 XNODE(I)=B1INV(1,I+1) 257
KKK=1 258
KR(1)=NMM 259
83001 IK=KB(KKK) 260
C
C DETERMINE WHETHER THE TIME TO REACH NODE IK IS NECESSARILY 261
C AS LARGE AS INDICATED FROM THE DUAL SOLUTION 262
C
C SMIN=999999. 263
ISMIN=0 264
DO B3000 I=1,M 265
IF(HEAD(I).NE.IK) GO TO 83000 266
SLACK=XNODF(HEAD(I))-XNODE(TAIL(I))-CTIME(I) 267
IF(SLACK.GE.SMIN) GO TO 83000 268
SMIN=SLACK 269
ISMIN=I 270
83000 CONTINUE 271
IF(SMIN.LT.0.0001) GO TO 83003 272
C
C THE TIME FOR NODE IK WAS UNNECESSARILY LARGE 273
C
XNODE(IK)=XNODE(IK)-SMIN 274
KKK=KKK-1 275
GO TO 83001 276
83003 IBB(KKK)=ISMIN 277
KKK=KKK+1 278
KB(KKK)=TAIL(ISMIN) 279
IF(TAIL(ISMIN).GT.1) GO TO 83001 280
KKR=KKK-1 281
IF(INDEXL.EQ.1) GO TO 3121 282
IPARM=IPARM+1 283
IF(IPARM.GT.4) GO TO 2910 284
IF(IPARM.EQ.3) GO TO 6400 285
IF(IPARM.EQ.4) GO TO 6401 286
C
C ICRTIP(L)= THE L-TH ACTIVITY ON THE ORIGINAL CRITICAL PATH 287
C

```

C KCPB= THE NUMBER OF ACTIVITIES ON THE ORIGINAL CRITICAL PATH
C
C TOTAL = BIINV(1,NMMP1)
C KCPB=KKB
C ICFITN(1) = NMM
C DO 2802 I=1,KCPB
C ICRITN(I+1) = KA(I+1)
2802 ICRITP(I)=IBB(I)
C X=TOTAL
C WRITE(6,851) X
851 FORMAT(1H0,5X,'THE CRITICAL PATH TIME WHEN EACH ACTIVITY''S COMPLETION TIME IS SET EQUAL TO ITS AVERAGE IS = ',E15.5)
C WRITE(6,7606) KKK
7606 FORMAT(1H0,10X,'THE ',I3,' NODES ON THE CRITICAL PATH ARE AS FOLLOWS BEGINNING WITH THE TERMINAL NODE:')
C *WS WRITE(6,7707) (KB(I),I=1,KKK)
7707 FORMAT(15X,20(I3,','))
C WRITE(6,7710) KKB
7710 FORMAT(1H0,10X,'THE ',I3,' CRITICAL ACTIVITIES ARE AS FOLLOWS BEGINNING WITH THE TERMINAL ACTIVITY:')
C *WS WRITE(6,7707) (IBB(I),I=1,KKB)
C READ(5,2920) THETA,LAMBDA
2920 FORMAT(2F5.2)
C WRITE(6,3071) THETA,LAMBDA
3071 FORMAT(1H0,10X,'THETA = ',E15.5,' LAMBDA = ',E15.5)

C
C SAMSIZ = THE NUMBER OF ACTIVITY TIME CONFIGURATIONS TO BE RANDOMLY SELECTED FOR CONSIDERATION IN EACH CLUSTER
C
C NOTE: SINCE THIS IS A RANDOM SAMPLE, SOME PERCENTILE COMBINATIONS MAY BE CONSIDERED MORE THAN ONCE.
C
C READ(5,3209) SAMSIZ
3209 FORMAT(1I0)

C
C THE COMPLETION TIME FOR ALL ACTIVITIES IS SET TO THEIR LOWER PERCENTILE. THE RESULTING CRITICAL PATH TIME IS A LOWER BOUND ON THE EXPECTED CRITICAL PATH TIME.
C
C DO 6402 I=1,M
6402 CTIME(I) = FLO(I)
CALL BINVA(62800)
CPLB= BIINV(1,NMMP1)
WRITE(6,6405) CPLB
6405 FORMAT(1H0,5X,'A LOWER BOUND ON THE EXPECTED CRITICAL PATH TIME IS '
* = ',F15.5)
WRITE(6,7606) KKK
WRITE(6,7707) (KB(I),I=1,KKK)
WRITE(6,7710) KKB
WRITE(6,7707) (IBB(I),I=1,KKB)

C
C THE COMPLETION TIME FOR ALL ACTIVITIES IS SET TO THEIR UPPER PERCENTILE. THE RESULTING CRITICAL PATH TIME IS A UPPER BOUND ON THE EXPECTED CRITICAL PATH TIME.
C
C DO 6406 I=1,M
6406 CTIME(I) = FHI(I)
CALL BINVA(62800)
CPUH= BIINV(1,NMMP1)
WRITE(6,6407) CPUH

```

```

6409 FORMAT(1H0,5X,'A UPPER BOUND ON THE EXPECTED CRITICAL PATH TIME IS      360
* = *,F15.5)          361
  WRITE(6,7626) KKK        362
  WRITE(6,7707) (KB(I),I=1,KKK)      363
  WRITE(6,7710) KKA        364
  WRITE(6,7707) (IBB(I),I=1,KKB)      365
C
C   FD(I) = THE LOWER BOUND ON THE EXPECTED CRITICAL PATH TIME      366
C   PLUS 1/IEDF OF THE DISTANCE TO THE UPPER BOUND      367
C   NLFFD(I) = THE OBSERVED NUMBER OF CRITICAL PATH TIMES      368
C   THAT ARE < OR= FD(I)      369
C   FD AND NLFFD ARE USED TO BUILD AN 'EMPIRICAL' DISTRIBUTION OF      370
C   THE CRITICAL PATH TIMES      371
C
C   C=(CPUB-CPLB)/IEDF      372
C   DO 6412 K=1,KCPB      373
C   DO 6412 I=1,IEDF      374
C   FD(I)=CPLB+I*C      375
6412 NLEFD(I) = 0      376
C
C   THE ASSOCIATE GROUPS ARE NOW FORMED      377
C
C   WRITE(6,3165)      378
3165 FORMAT(1H1+5X,'THE ASSOCIATES ARE NOW IDENTIFIED:')      379
  I!!!!=1      380
  DO 2825 I=1,M      381
2825 CTIME(I)=COT(I)      382
  IWWWQ=ICRITP(I)      383
  CHANG= LAMBDA*SIGMA(IWWWQ)      384
  TFX=COT(IWWWQ)-CHANG      385
  IF(TFX.LT.0.0) CHANG=COT(IWWWQ)      386
  CTIME(IWWWQ)=COT(IWWWQ)-CHANG      387
  CALL RINVA(82800)      388
2801 CONTINUE      389
  IF (ISTAT(IWWWQ).EQ.1) CALL BINV1(822825,COT(IWWWQ),CTIME(IWWWQ),*
*IWWWQ)      390
  REDCOS(IWWWQ) = REDCOS(IWWWQ)+COT(IWWWQ)-CTIME(IWWWQ)      391
22825 CTIMF(IWWWQ)=COT(IWWWQ)      392
  IWWWQ=ICRITP(I!!!!)      393
  CHANG= LAMBDA*SIGMA(IWWWQ)      394
  TEX=COT(IWWWQ)-CHANG      395
  IF(TEX.LT.0.0) CHANG=COT(IWWWQ)      396
  CTIME(IWWWQ)=COT(IWWWQ)-CHANG      397
  IF (ISTAT(IWWWQ).EQ.1) CALL BINV1(822800,CTIME(IWWWQ),COT(IWWWQ),*
*IWWWQ)      398
  REDCOS(IWWWQ)=REDCOS(IWWWQ)-COT(IWWWQ)+CTIME(IWWWQ)      399
  GO TO 22800      400
C
C   DETERMINE ASSOCIATE GROUP      401
C
2910 NINAG(I!!!!)=0      402
  DO 2911 K=1,KKB      403
  KK=1      404
-2913 IF(IBB(K).EQ.ICRITP(KK)) GO TO 2911      405
  IF(KK.GE.KCPB) GO TO 2912      406
  KK=KK+1      407
  GO TO 2913      408
2912 NINAG(I!!!!)=NINAG(I!!!!)+1      409
  ASSGRD(I!!!!,NINAG(I!!!!))=IBB(K)      410
2911 CONTINUE      411

```

```

      WRITE(6,2915) I||||,ICRITP(|||),NINAG(|||)

2915  FORMAT(1HO,10X,'THE NUMBER OF ASSOCIATES ASSOCIATED WITH THE ',I3,
     *'-TH CRITICAL PATH ACTIVITY, I.E. ACTIVITY ',I3,', IS = ',I3)
     IDUCK=NINAG(|||)

     IF(IDUCK,FQ,0) GO TO 2810
     WRITE(6,2916) (ASSGRP(|||,I),I=1,1DUCK)
2916  FORMAT(1HO,15X,'THE ACTIVITIES IN THE ASSOCIATE GROUP ARE AS FOLLO
     *WS'//,15X,59(I3,''))
2810  I||||=||||+1
     IF(|||,LF,KCPR) GO TO 2801

```

C
C DETERMINE THE CLUSTERS
C

THE CLUSTERS ARE POOLED TOWARD THE TERMINAL NODE
NCLUS = THE NUMBER OF NCN-EMPTY CLUSTERS
NINCL(I) = THE NUMBER OF ACTIVITIES IN THE I-TH CLUSTER
INCLUS(I,J) = THE J-TH ACTIVITY IN THE I-TH CLUSTER
NCLINC(I) = THE NUMBER OF CLUSTERS COMPRISING THE I-TH
CLUSTER AFTER POOLING
CLINCL(I,J) = THE J-TH CLUSTER WHICH HAS BEEN POOLED INTO
THE I-TH CLUSTER

NCLINC AND CLINCL HELP KEEP TRACK OF WHICH CLUSTER THE
CRITICAL PATH ACTIVITIES ARE IN

BELLOW FORMS CLUSTERS BY PUTTING EACH CRITICAL PATH ACTIVITY IN
SEPARATE CLUSTER AND THEN ADDING EACH CRITICAL PATH ACTIVITY'S
ASSOCIATES TO ITS CLUSTER

```

NCLUS=KCPR
DO 3020 I=1,KCPR
NCLINC(I)=1
CLINCL(I,1)=I
NINCL(I)=NINAG(I)+1
INCLUS(I,1)=ICRITP(I)
IF(NINAG(I),EQ,0) GO TO 3020
IDUCK=NINCL(I)
DO 3021 J=2,1DUCK
JJ=J-1
INCLUS(I,J)=ASSGRP(I,JJ)
CONTINUE

```

C
C BELOW POOLS CLUSTERS FORMED FROM ASSOCIATES
C

```

IA=0
IA=IA+1
IF(IA,GE,KCPR) GO TO 3030
IF(NCLUS,EQ,1) GO TO 3030
IDIA=NINCL(IA)
IF(IDIA,FQ,0) GO TO 3031
IAA=IA+1
DO 3023 II=IAA,KCPR
IDII=NINCL(II)
IF(IDII,EQ,0) GO TO 3023
DO 3025 I=1,1DIA
DO 3025 J=1,1DII
IF(INCLUS(II,J),EQ,INCLUS(IA,I)) GO TO 3027
CONTINUE
GO TO 3023

```

*COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION*

420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479

3027 NCLUS=NCLUS-1 480
 DO 3028 J=1,IDL
 DO 3029 I=1,IDA
 IF(NINCL(I,J),EQ,INCLUS(IA,I)) GO TO 3028 481
 3029 CONTINUE 482
 NINCL(IA)=NINCL(IA)+1 483
 INCLUS(IA,NINCL(IA))=INCLUS(I,J) 484
 3028 CONTINUE 485
 NINCL(IJ)=0 486
 NCLINC(IA)=NCLINC(IA)+1 487
 CLINCL(IA,NCLINC(IA))=II 488
 NCLINC(IA)=0 489
 3023 CONTINUE 490
 GO TO 3031 491
 3030 CONTINUE 492
 C 493
 C BELOW DESCRIBES CLUSTERS AFTER POOLING BASED ON THE ASSOCIATES 494
 C 495
 WRITE(6,3033) NCLUS 496
 3033 FORMAT(1H1,10X,'THERE ARE ',I3,' NONEMPTY CLUSTERS AFTER POOLING ON 497
 *N THE BASIS OF ASSOCIATES ONLY.') 498
 II=0 499
 DO 3034 I=1,KCPB 500
 IF(NINCL(I),EQ,0) GO TO 3034 501
 II=II+1 502
 IDUCJ=NINCL(I) 503
 WRITE(6,3035) I,(INCLUS(I,J),J=1,IDUCJ) 504
 3035 FORMAT(1H0,10X,'THE ACTIVITIES IN THE ',I3,'-TH CLUSTER ARE AS FOL 505
 *LOWS:./,15X,50(I3,'')) 506
 3034 CONTINUE 507
 C 508
 C DESCRIBES WHERE EACH ACTIVITY IS BEFORE ELIMINANTS ARE 509
 C CONSIDERED 510
 C 511
 C EXAMINE EACH ACTIVITY AND DETERMINE WHICH CLUSTER, IF ANY, IT IS 512
 C IN. 513
 C LFFT(I)=0 IMPLIES THAT THE I-TH ACTIVITY IS NOT IN ANY 514
 C CLUSTER 515
 C LEFT(I)=J IMPLIES THE I-TH ACTIVITY IS IN THE J-TH 516
 C CLUSTER 517
 C 518
 WRITE(6,3104) 519
 3104 FORMAT(1H0,10X,'THE CLUSTER TO WHICH EACH ACTIVITY BELONGS:./,15X 520
 *'(ZERO IMPLIES THAT THE ACTIVITY IS NOT IN ANY CLUSTER)') 521
 DO 3101 I=1,M 522
 LFFT(I)=0 523
 DO 3102 J=1,KCPB 524
 IF(NINCL(J),EQ,0) GO TO 3102 525
 IDUCK=NINCL(J) 526
 DO 3110 K=1,IDUCK 527
 IF(I,EQ,INCLUS(J,K)) GO TO 3107 528
 3110 CONTINUE 529
 3102 CONTINUE 530
 GO TO 3101 531
 3107 LFFT(I)=J 532
 3101 WRITE(6,3103) I,LEFT(I) 533
 3103 FORMAT(1H ,15X,'THE ',I3,'-TH ACTIVITY IS IN THE ',I3,'-TH CLUSTER 534
 *') 535
 INDEXL=1 536
 537
 538
 539

COPY AVAILABLE TO DDC DOES NOT
 PERMIT FULLY LEGIBLE PRODUCTION

LEFTOVERS ARE ACTIVITIES NOT IN CLUSTERS AFTER ASSOCIATES HAVE BEEN CONSIDERED BUT BEFORE ELIMINANTS HAVE BEEN CONSIDERED

DETERMINE THE NUMBER OF LEFTOVERS. NLEFT
 $LFTD(L) = J$ IMPLIES THAT THE L-TH LEFTOVER IS THE J-TH ACTIVITY

```
NLEFT=0
DO 3122 J=1,M
IF(LFTD(J).NE.0) GO TO 3122
NLEFT=NLEFT+1
LFTD(NLEFT)=J
3122 CONTINUE
WRITE(6,3123) NLEFT
3123 FORMAT(1H0,10X,'THERE ARE ',I3,' ACTIVITIES NOT IN ANY CLUSTER YE
*T.')
WRITE(6,3323)
3323 FORMAT(1H1,5X,'THE ELIMINANTS OF EACH NON-CRITICAL-PATH ACTIVITY A
*RE NOW DETERMINED!')
```

ELIMINANTS FOR EACH NON-CRITICAL-PATH ACTIVITY ARE NOW DETERMINED

NNCP = THE NUMBER OF ACTIVITIES NOT ON THE CRITICAL PATH
 $NONCP(LF) =$ THE LE-TH ACTIVITY NOT ON THE CRITICAL PATH

```
NNCP=M-KCP
LF=0
DO 5000 I=1,M
J=1
5001 IF(I.EQ.1.CRITP(J)) GO TO 5000
J=J+1
IF(J.LF.KCP) GO TO 5001
5002 LF=LF+1
NONCP(LF)=I
5003 CONTINUE
WRITE(6,5005) NNCP
5005 FORMAT(1H0,0SX,'THERE ARE ',I3,' ACTIVITIES NOT ON THE CRITICAL PA
*TH. THEY ARE AS FOLLOWS:')
IF(NNCP.EQ.0) GO TO 3124
DO 5006 I=1,LE
```

COPY AVAILABLE TO DDG DOES NOT PERMIT FULLY LEGIBLE PRODUCTION

```
5006 WRITE(6,5007) I,NONCP(I)
5007 FORMAT(1H ,15X,I3,' ')
IF(NNCP.EQ.0) GO TO 3124
LF=0
3126 LE=LF+1
IF (ISTAT(IWWWO).EQ.1) CALL RINV1(623127,COT(IWWWO),CTIME(IWWWO),
*IWWWO)
RFDCOS(IWWWO) = RFDCOS(IWWWO)-CTIME(IWWWO)+COT(IWWWO)
3127 CONTINUE
CTIME(IWWWO) = COT(IWWWO)
CTIME(NONCP(LE)) = COT(NONCP(LE)) + THETA*SIGMA(NONCP(LE))
IF (ISTAT(NONCP(LE)).EQ.1) CALL RINV1(67756,CTIME(NONCP(LE)),
* COT(NONCP(LE)),NONCP(LE))
RFDCOS(NONCP(LE))=RFDCOS(NONCP(LE))-COT(NONCP(LE))+CTIME(NONCP(LE))
*)
7756 IWWWO = NONCP(LE)
WRITE(6,3152) NONCP(LE),CTIME(NONCP(LE))
3152 FORMAT(1H0,///, 5X,'THE COMPLETION TIME FOR THE ',I3,'-TH ACTIVITY
```

* HAS BEEN CHANGED TO ',E15.5)

GO TO 22800

3121 CONTINUE

C DETERMINE THE ELIMINANTS OF THE LE-TH ACTIVITY NOT ON THE
CRITICAL PATH

C NE = THE NUMBER OF ELIMINANTS FOR THE LE-TH
ACTIVITY NOT ON THE CRITICAL PATH

C EGRP(J) = THE J-TH ELIMINANT FOR THE LE-TH ACTIVITY
NOT ON THE CRITICAL PATH

C

NE=0

DO 3130 K=1,KCPB

DO 3131 T=1,KKB

IF(TBB(I).EQ.1)CRITP(K)) GO TO 3130

3131 CONTINUE

NE=NE+1

EGRP(NE)=ICRITP(K)

3130 CONTINUE

WRITE(6,3133) NE,NONCP(LE)

3133 FORMAT(1H0,10X,'THERE ARE ',I3,' ELIMINANTS CORRESPONDING TO ACTIV
*ITY ',I3)

IF(NE.EQ.0) GO TO 3171

DO 3135 K=1,NE

3135 WRITE(6,3136) K,NONCP(LE),EGRP(K)

3136 FORMAT(1H ,14X,'THE ',I3,'-TH ELIMINANT CORRESPONDING TO ACTIVITY
*',I3,' IS ACTIVITY ',I3)

C DETERMINE WHETHER NONCP(LF) IS AN ASSOCIATE

C JA = 1 IF NONCP(LE) IS AN ASSOCIATE

C JA = 2 IF NONCP(LE) IS NOT AN ASSOCIATE

C

K=NONCP(LE)

JA=1

IF(LEFT(K).EQ.0) JA=2

IF(JA.EQ.2) GO TO 501C

IT=LEFT(K)

C THE IT-TH CLUSTER IS EXPANDED TO INCLUDE ELIMINANTS

C

GO TO 5011

5010 CONTINUE

C ITTT IS THE ACTIVITY NUMBER OF THE FIRST ELIMINANT

C IT IS THE CLUSTER TO WHICH THE FIRST ELIMINANT CURRENTLY BELONG

C

ITTT=EGRP(1)

IT=LEFT(ITT)

LEFT(NONCP(LF))=IT

C THE IT-TH CLUSTER IS EXPANDED TO INCLUDE ELIMINANTS

C

NINCL(IT)=NINCL(IT)+1

INCLUS(IT,NINCL(IT))=NONCP(LE)

IF(NE.EQ.1) GO TO 3171

5011 DO 3172 J=JA,NE

C IU IS THE ACTIVITY NUMBER OF THE NEXT ELIMINANT

C IF IU IS IN CLUSTER K, THEN CLUSTER K IS POOLED INTO CLUSTER IT

```

IU=EGRP(J)                                660
K=LEFT(IU)                                 661
3182 IF(IT.EQ.K) GO TO 3172               662
NCLUS=NCLUS-1                             663
IW=NCLINC(K)                            664
DO 3183 IA=1,IW                           665
LFFT(ICR1TP(CLINCL(K,IA)))=IT          666
NCLINC(IT)=NCLINC(IT)+1                 667
3183 CLINCL(IT,NCLINC(IT))=CLINCL(K,IA) 668
NCLINC(K)=n                               669
IW=NINCL(K)                            670
NINCL(K)=0                               671
DO 3184 IA=1,IW                           672
LEFT(INCLUS(K,IA))=IT                  673
NINCL(IT)=NINCL(IT)+1                 674
3184 INCLUS(IT,NINCL(IT))=INCLUS(K,IA) 675
3172 CONTINUE                            676
3171 CONTINUE                            677
IF(LE.LT.NNNCP) GO TO 3126              678
C                                         679
C           END OF POOLING BASED ON ELIMINANTS EXCEPT FOR THE FOLLOWING 680
C           DESCRIPTION                                         681
C                                         682
WRITE(6,3173) NCLUS                      683
3173 FORMAT(1H1,05X,'THERE ARE ',I3,' CLUSTERS.') 684
DO 3176 I=1,KCPB                         685
IF(NINCL(I).EQ.0) GO TO 3176             686
IDD=NINCL(I)                            687
WRITE(6,3174) NINCL(I),I,(INCLUS(I,J),J=1,IDD) 688
3174 FORMAT(1H0,10X,'THERE ARE ',I3,' ACTIVITIES IN THE ',I3,'-TH CLUST 689
*ER. THEY ARE AS FOLLOWS:',/,20X,50(I3,',')) 690
IDUCK=NCLINC(I)                         691
WRITE(6,3175) NCLINC(I),(CLINCL(I,J),J=1,IDUCK) 692
3175 FORMAT(1H0,15X,I3,' CLUSTERS HAVE BEEN POOLED TO MAKE THIS CLUSTER 693
*. THEY WERE AS FOLLOWS:',/,20X,50(I3,',')) 694
3176 CONTINUE                            695
WRITE(6,2776) SAMSIZ                     696
2776 FORMAT(//5X, ' THE NUMBER OF PERCENTILE COMBINATIONS EXP 697
*LICITLY CONSIDERED IN DETERMINING THE UPPER BOUNDS AND LOWER BOUND 698
*S*/,6X, 'ON THE NETWORK COMPLETION TIME DISTRIBUTION AND THE UPPER 699
*ROUNDS ON ITS MOMENTS IS EQUAL TO ',I5) 700
CALL CLOCK(XRAN)
IYUTS = XRAN
WRITE(6,3238) IYUTS
3238 FORMAT(1H0,5X,'THE INITIALIZATION PARAMETER FOR THE SAMPLING IS I 704
*Y = ',I10)                            705
C                                         706
C           STATEMENT NUMBER 3124 MARKS THE END OF POOLING CLUSTERS BASED 707
C           ON LEFTOVERS AND ELIMINANTS                                         708
C                                         709
3124 CONTINUE                            710
C                                         711
C                                         712
C           THE FINAL CLUSTERS HAVE NOW BEEN DETERMINED 713
C           THE 2**NINCL(I) RUNS ARE NOW AVERAGED FOR ALL I WITH NINCL(I)>0 714
C           THE IZ(I) ARE USED TO REPRESENT ALL OF THE 2**NINCL 715
C           POSSIBILITIES.                                         716
C           THIS IS WHERE THE BINARY REPRESENTATION IS CONSTRUCTED. 717
C                                         718
6008 IF(ICBCP.EQ.1) ICBCP=2            719

```

```

INDEXL=2                                720
IR=0                                     721
3200 IR=IR+1                            722
DO 6031 I=1,10                           723
6031 MOMEFT(IR,I) = 0.0                 724
IF(IR.GT.KCPB) GO TO 3208               725
IF(NINCL(IR).EQ.0) GO TO 3200           726
IDUCK=NCLINC(IR)                       727
DO 3410 I=1,1DUCK                      728
K=CCLINCL(IR,I)                        729
3410 CONTINUE                           730
L=ICRITP(K)                           731
3310 IP = 0                             732
NIR = 0                               733
C
C      NIB   = NUMBER OF PERCENTILE COMBINATIONS IN THE SAMPLE    734
C      PNIB  = PERCFNTAGE OF THE TOTAL NUMBER OF PERCENTILE COMBINATIONS 735
C                  EXPLICITLY CONSIDERED 736
C      NSAVF = VECTOR CONTAINING THE LOWER BOUNDS ON THE NETWORK    737
C                  COMPLETION TIME DISTRIBUTION TO BE AVERAGED WITH THE 738
C                  UPPER BOUNDS ON THE NETWORK TO YIELD THE AVERAGE NETWORK 739
C                  COMPLETION TIME DISTRIBUTION. 740
C      KNIR  = NIR ASSOCIATED WITH NSAVE. 741
C
C
C      IC=NINCL(IR)                         742
C      IB=2**NINCL(IR)                     743
C      IDOALL = 1 MEANS ALL PERCENTILE COMBINATIONS ARE EXPLICITLY 744
C                  CONSIDERED. 745
C      IDOALL = 0 MEANS TO SAMPLE. 746
C
C      IDCALL = 0                           747
C      IF (SAMSIZ.LE.0 .OR. SAMSIZ.GE.IB) IDOALL=1 748
C      DO 3222 I=1,M 749
3222 CTIME(I) = COT(I) 750
GO TO 20000 751
C
C      STATEMENT 3204 IS THE RE-ENTRY POINT FROM THE DUAL SIMPLEX 752
C                  ALGORITHM WHEN A CLUSTER AVERAGE IS BEING COMPUTED 753
C
3204 CONTINUE 754
IF (IDLB.EQ.1) GO TO 990C 755
DO 6032 I=1,10 756
6032 MOMEFT(IR,I) = MOMEFT(IR,I) + B1INV(1,NMMP1)**I 757
IF (ICBCP.NE.2) GO TO 10001 758
9900 X= B1INV(1,NMMP1) 759
X=X-1.0D-10 760
I=0 761
6420 I=I+1 762
IF(X.GT.FD(I)) GO TO 6420 763
NLEFD(I) = NLEFD(I) + 1 764
10001 CONTINUE 765
20000 IP=IP+1 766
NIR = NIR+1 767
C
C      GENERATE NEXT PERCENTILE COMBINATIONS TO BE EXPLICITLY 768
C                  CONSIDERED. 769
C
C      IF (IDOALL.EQ.0) GO TO 20500 770

```

```

IF (IP.GT.IB) GO TO 3207          780
RANSAM = IP                      781
GO TO 20501                      782
20500 IF (NIB.GT.SAMSIZ) GO TO 3207 783
IYUTS = IYUTS*65539               784
IF (IYUTS) 6210,6211,6211         785
6210 IYUTS = IYUTS+2147483647+1   786
6211 XRAN = IYUTS                787
XRAN = XRAN*.4656613F-9          788
RANSAM = XRAN*DFLOAT(IB-1) + 1    789
20501 CONTINUE                   790
C                                     791
C     CONVERT THE RANDOM NUMBER, RANSAM, TO A BINARY NUMBER TO DEFINE A 792
C     PERCENTILE COMBINATION.                                         793
C                                     794
KRAM = RANSAM                    795
DO 8505 I=1,1C                  796
IHALF = KRAM/2                  797
IZ = KRAM - IHALF*2             798
L = INCLUS(IP,I)               799
CTIMF(L) = IZ*FHT(L)-IZ*FL0(L) + FL0(L)                         800
8505 KRAM = IHALF              801
CALL BINVA(82800)               802
3207 NIB = NIB-1                 803
IF (IDLB.F0.1) GO TO 9011       804
DO 6030 I=1,10                  805
6030 MOMENT(IR,I) = MOMENT(IR,I)/NIB                            806
IF (ICBCP.E0.2) GO TO 6011       807
GO TO 3200                      808
3208 WRITE (6,6362)              809
6362 FORMAT (1H1)                810
PNTB = DFLOAT(NIB*100)/DFLOAT(IB)                                811
IF (SAMSIZ.LF.0) WRITE(6,6045)                               812
IF (SAMSIZ.GT.0) WRITE(6,6055) SAMSIZ                           813
6056 FORMAT (5X,' THE FOLLOWING TABLE WAS COMPUTED CONSIDERING AT MOST 814
*,18,' PERCENTILE COMBINATIONS ')                           815
6046 FORMAT (5X,' THE FOLLOWING TABLE WAS COMPUTED CONSIDERING ONLY ', 816
*,18,' PERCENTILE COMBINATIONS OR ',F6.2,' PERCENT OF ALL COMBINA 817
*TIONS.')                                         818
6045 FORMAT (5X,' THE FOLLOWING TABLE WAS COMPUTED CONSIDERING ALL PERC 819
*ENTILE COMBINATIONS.')                                820
DO 10000 J=1,10                     821
ITMAX=0                           822
TMAX=0                            823
DO 5021 I=1,KCPB                 824
IF(NINCL(I).FQ.0) GO TO 5021      825
IF (MOMENT(I,J).LE.TMAX) GO TO 5021 826
TMAX = MEMENT(I,J)                827
ITMAX=I                           828
5021 CONTINUE                      829
WRITE (6,5023) J,J,MOMENT(ITMAX,J)                         830
5023 FORMAT (1H0,5X,'A LOWER BOUND. T-(*,I2,';THETA,LAMDA), ON THE ' 831
*,           I2 ,'-TH MOMENT OF THE NETWORK COMPLETION TIME = ',F15.5) 832
10000 CONTINUE                      833
C                                     834
C     BEGIN THE PROCEDURE FOR DETERMINING UPPER BOUNDS ON THE MOMENTS 835
C     OF THE NETWORK COMPLETION TIME AND LOWER BOUNDS ON THE            836
C     DISTRIBUTION OF THE COMPLETION TIMES.                           837
C                                     838
C                                     839

```

C POOL ALL OF THE PREVIOUS CLUSTERS INTO ONE CLUSTER 840
C NCL = THE INDEX OF THE RESULTANT POOLED CLUSTER 841
C NNCL = THE NUMBER OF ACTIVITIES IN THIS POOLED CLUSTER 842
C 843
C IF(NCLUS.GT.1) GO TO 6000 844
I=0 845
6001 I=I+1 846
IF(NINCL(I).EQ.0) GO TO 6001 847
NCL=I 848
GO TO 6002 849
6000 NMAX = 0 850
DO 6003 I=1,KCPB 851
IF(NINCL(I).LE.NMAX) GO TO 6003 852
NCL =I 853
NMAX=NINCL(I) 854
6003 CONTINUE 855
DO 6004 I=1,KCPB 856
IF(NINCL(I).EQ.0) GO TO 6004 857
IF(I.GE.NCL) GO TO 6004 858
K=NINCL(NCL) 859
JJ=NINCL(I) 860
NINCL(I)=0 861
DO 6005 J=1,JJ 862
K=K+1 863
6005 INCLUS(NCL,K)=INCLUS(I,J) 864
NINCL(NCL)=NINCL(NCL)+JJ 865
6004 CONTINUE 866
6002 NNCL=NINCL(NCL) 867
C 868
C 869
C 870
C THE UPPER BOUNDS, T+(R,THETA,LAMBDA), ARE NOW DETERMINED. 871
C 872
C 873
C FOR THE SAKE OF NUMERICAL ACCURACY THE PERT PROBLEM 874
C WITH NEW ACTIVITY TIMES IS INITIALLY SOLVED FROM SCRATCH 875
C INSTEAD OF UPDATING AN OLD SOLUTION. AFTER THIS 876
C REINITIALIZATION, THE REMAINING CRITICAL PATH TIMES ARE 877
C DETERMINED BY UPDATING THIS SOLUTION. 878
C 879
DO 6006 I=1,N 880
CTIME(I)=FHI(I) 881
6006 COT(I)=FHI(I) 882
DO 6007 J=1,NNCL 883
I =INCLUS(NCL,J) 884
6007 CTIME(I)=.5*(FLO(I)+FHI(I)) 885
DO 7101 I=1,IEDF 886
7101 NLEFD(I) = 0 887
ICRCP=1 888
GO TO 6010 889
6011 WRITE (6,6264) 890
6264 FORMAT(///)
PNIB = DFLDAT(NIB*100)/DFLDT(1B) 891
IF (1DOALL.EQ.0) WRITE(6,6046) NIB,PNIB 892
IF (1DOALL.EQ.1) WRITE(6,6045) 893
DO 10009 J=1,10 894
10009 WRITE (6,6012) J,J,MOMENT(NCL,J) 895
6012 FORMAT (1H0,5X,'AN UPPER BOUND, T+',I2,';THETA,LAMBDA), ON THE ' 896
*,I2,'-TH MOMENT OF THE NETWORK COMPLETION TIME = ',E15.5) 897
DO 6900 I=2,IEDF 898
6900

```

II=I-1
6900 NLEFD(I) = NLEFD(I) + NLEFD(II)
      WRITE (6,6362)
      PNIB = DFL0AT(NIB*100)/DFL0AT(IB)
      IF (ID0ALL.EQ.0) WRITE(6,6046) NIB,PNIB
      IF (ID0ALL.EQ.1) WRITE(6,6045)
      WRITE(6,6423)
5423 FORMAT(1H0,5X, 'A LOWER BOUND ON THE NETWORK COMPLEFTION TIME DISTR
*IBUTION: F-(.;THETA;LAMBDA)')
      KNIB = NIB
      DO 6421 I=1,IEDF
      NSAVE(I)=NLEFD(I)
      X=NLEFD(I)
      X=X/NIB
      X=X/NIB
6421 WRITE(6,6422) FD(I),THETA,LAMBDA,X
6422 FORMAT(17X,'F-(',E15.5,';',E15.5,';',E15.5,') = ',E15.5)
C
C
C      BEGIN THE PROCEDURE FOR DETERMINING UPPER BOUNDS ON THF
C      NETWORK COMPLETION TIME DISTRIBUTION
C
C      FOR THE SAKE OF NUMERICAL ACCURACY THE PERT PROBLEM
C      WITH NEW ACTIVITY TIMES IS INITIALLY SOLVED FROM SCRATCH
C      INSTEAD OF UPDATING AN OLD SOLUTION. AFTER THIS
C      REINITIALIZATION, THE REMAINING CRITICAL PATH TIMES ARE
C      DETERMINED BY UPDATING THIS SOLUTION.
      DO 9006 I=1,M
      CTIME(I)=FL0(I)
9006 CNT(I)=FL0(I)
      DO 9007 J=1,NNCL
      I =INCLUS(NCL,J)
9007 CTIME(I)=.5*(FL0(I)+FHI(I ))
      DO 9101 I=1,IFDF
9101 NLFFD(I) = 0
      ICBCP = 1
      IDLB = 1
      GO TO 6010
9011 DO 9990 I=2,IEDF
      II=I-1
      NLEFD(I) = NLEFD(I) + NLEFD(II)
      WRITE (6,6254)
      PNIB = DFL0AT(NIB*100)/DFL0AT(IB)
      IF (ID0ALL.EQ.0) WRITE(6,6046) NIB,PNIB
      IF (ID0ALL.EQ.1) WRITE(6,6045)
      WRITE(6,6423)
9423 FORMAT(1H0,5X, 'AN UPPER BOUND ON THE NETWORK COMPLETION TIME DISTR
*IBUTION: F+(.;THFTA;LAMBDA)')
      DO 9421 I=1,IEDF
      X=NLEFD(I)
      X=X/NIB
9421 WRITE(6,6422) FD(I),THETA,LAMBDA,X
9422 FORMAT(17X,'F+(',E15.5,';',E15.5,';',E15.5,') = ',E15.5)
C
C      THE APPROXIMATE NETWORK COMPLETION TIME DISTRIBUTION
      WRITE (6,6362)
C
      WRITE(6,9472)
9472 FORMAT(1H0,5X, 'AN APPROXIMATE NETWORK COMPLETION TIME DISTRIBUTION
*:',//,15X,'F(.;THETA,LAMBDA) = .5 * ( F+(.;THETA,LAMBDA) + F-(.;TH
*ETA,LAMBDA) )',//)

```

```

DO 9471 I=1,IEDF          960
X = .5*DFLOAT(NSAVE(I))/KNIR+.5*DFLOAT(NLEFD(I))/NIR 961
9471 WRITE(6,9473) FD(I),THETA,LAMRDA,X                962
9473 FORMAT(17X,' F(*,E15.5,*;*,E15.5,*;*,E15.5,* ) = *,E15.5) 963
999 WRITE(6,850)           964
STOP               965
END               966
SUBROUTINE BINVA(*)      967
IMPLICIT RFAL*8 (A-H,O-Z) 968
COMMON B1INV,REDCOS,CTIME,XB1,INBASE,IHEAD,ITAIL,NMMP1,NMM,N,ISTAT 969
COMMON M,MPI             970
DIMENSION ISTAT(100),IHEAD(60),ITAIL(60),XB1(41)            971
DIMENSION B1INV(41,41),INBASE(40),CTIME(100),REDCOS(100)        972

```

C UPDATE THE FIRST ROW OF B1INV AFTER CHANGING CTIME 973

```

C DO 1 I=2,NMMP1          974
1 B1INV(1,I) = 0.0          975
DO 1 J=2,NMMP1          976
1 B1INV(1,I) = B1INV(1,I) + B1INV(J,I)*CTIME(INBASE(J-1)) 977

```

C UPDATE VALUE OF THE OBJECTIVE FUNCTION 978

```

C XB1(1) = B1INV(1,NMMP1) 979
C RETURN1                  980
C END                      981
C SUBROUTINE BINVI (*,TMNEW,TMOLD,1D) 982
C IMPLICIT RFAL*8 (A-H,O-Z) 983
C COMMON B1INV,REDCOS,CTIME,XB1,INBASE,IHEAD,ITAIL,NMMP1,NMM,N,ISTAT 984
C COMMON M,MPI              985
C DIMENSION ISTAT(100),IHEAD(60),ITAIL(60),XB1(41)            986
C DIMENSION B1INV(41,41),INBASE(40),CTIME(100),REDCOS(100)        987

```

C COMPUTE THE REDCOS CORRESPONDING TO ONE CHANGE IN CTIME 988

```

C DO 2 I=1,NMM          989
2 IF(INBASE(I).EQ.1D) II=I+1 990
DIFF = TMNEW-TMOLD          991
      TMNEW IS THE NEW TIME AND TMOLD IS THE OLD TIME CORRESPONDING 992
      TO THE SINGLE CHANGE IN CTIME 993

```

```

C DO 1 K=1,M          994
1 IF(ISTAT(K).EQ.1) GO TO 1 995
REDCOS(K) = REDCOS(K)-DIFF*(B1INV(II,IHEAD(K)+1) - 996
* B1INV(II,ITAIL(K)+1)) 997

```

```

1 CONTINUEF          998
DO 3 K=MP1,N          999
3 IF (ISTAT(K).EQ.1) GO TO 3 1000
REDCOS(K) = REDCOS(K) - DIFF*B1INV(II,K-M+1) 1001
3 CONTINUEF          1002

```

C UPDATE THE FIRST ROW OF B1INV AFTER CHANGING CTIME 1003

```

C DO 10 I=2,NMMP1         1004
10 B1INV(1,I)=B1INV(1,I)+DIFF*B1INV(II,I) 1005
10 CONTINUE          1006

```

C UPDATE VALUE OF THE OBJECTIVE FUNCTION 1007

```

C XB1(1) = B1INV(1,NMMP1) 1008

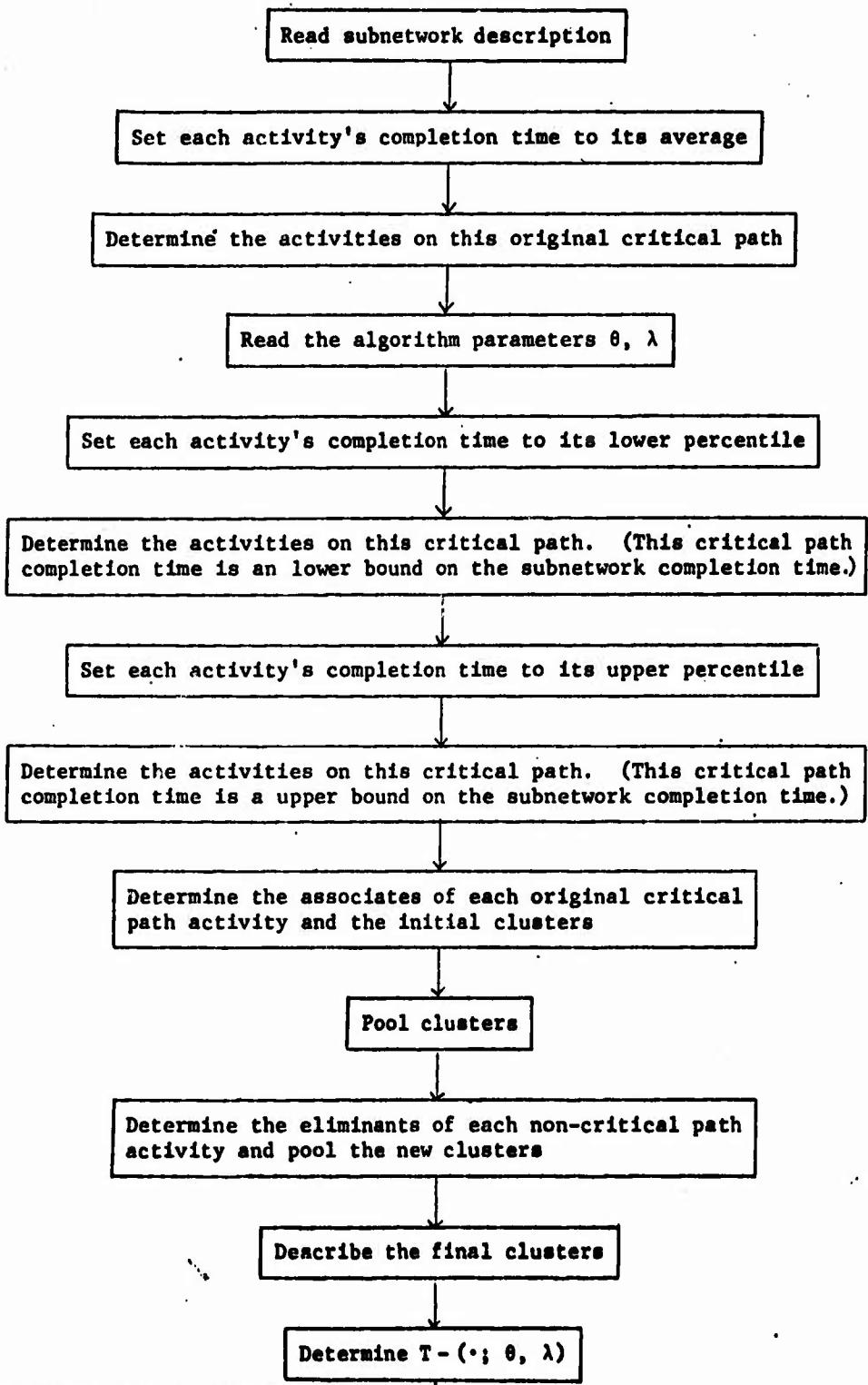
```

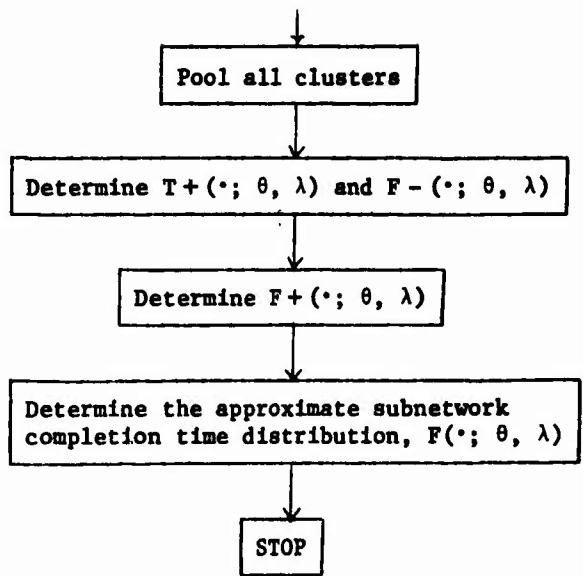
96

RETURN1
END

1020
1021

Original Subnetwork Analysis Program: Flowchart





APPENDIX D

Monte Carlo PERT Simulation Program

The Monte Carlo PERT Simulation Program will generate a random sample of network completion times. The required input is

- (a) an acyclic network with one sink,
- (b) parameters describing each activity's completion time distribution, and
- (c) the size of the random sample to be generated.

Currently, the program generates each activity's random sample of completion times from a chi-square distribution with 3 degrees of freedom that has been linearly transformed to have specified 15-th and 85-th percentiles. The activity time distribution can be easily changed. The basic output is the ordered sample of random network completion times and the corresponding empirical distribution function. The critical paths associated with the sample of network completion times are not determined.

The generation of the random sample of network completion times involves only one network but varying values of the individual activity completion times. It is computationally faster to find the network completion time for a new set of activity completion times by "updating" the network completion time for a previous set of activity completion times than it is to start all over each time. Since the Simplex Algorithm applied to the dual of the PERT problem is ideally suited for this type of "updating", the basic computational technique for determining the network completion times is the Simplex Algorithm (see e.g., G. Hadley, Linear Programming).

A listing of the Monte Carlo Simulation Program is given at the end of this appendix.

Specific Input Instructions:

- Card 1. Col. 1-3 : The number of activities in the network, Format (I3).
Col. 4-6 : The number of nodes in the network, Format (I3).
- Card 2. Col. 11-15: The number of random network completion times to be generated, Format (I5).
Col. 21-25: The number of parameters needed for generating the random activity completion times, Format (15).

For each activity one card with:

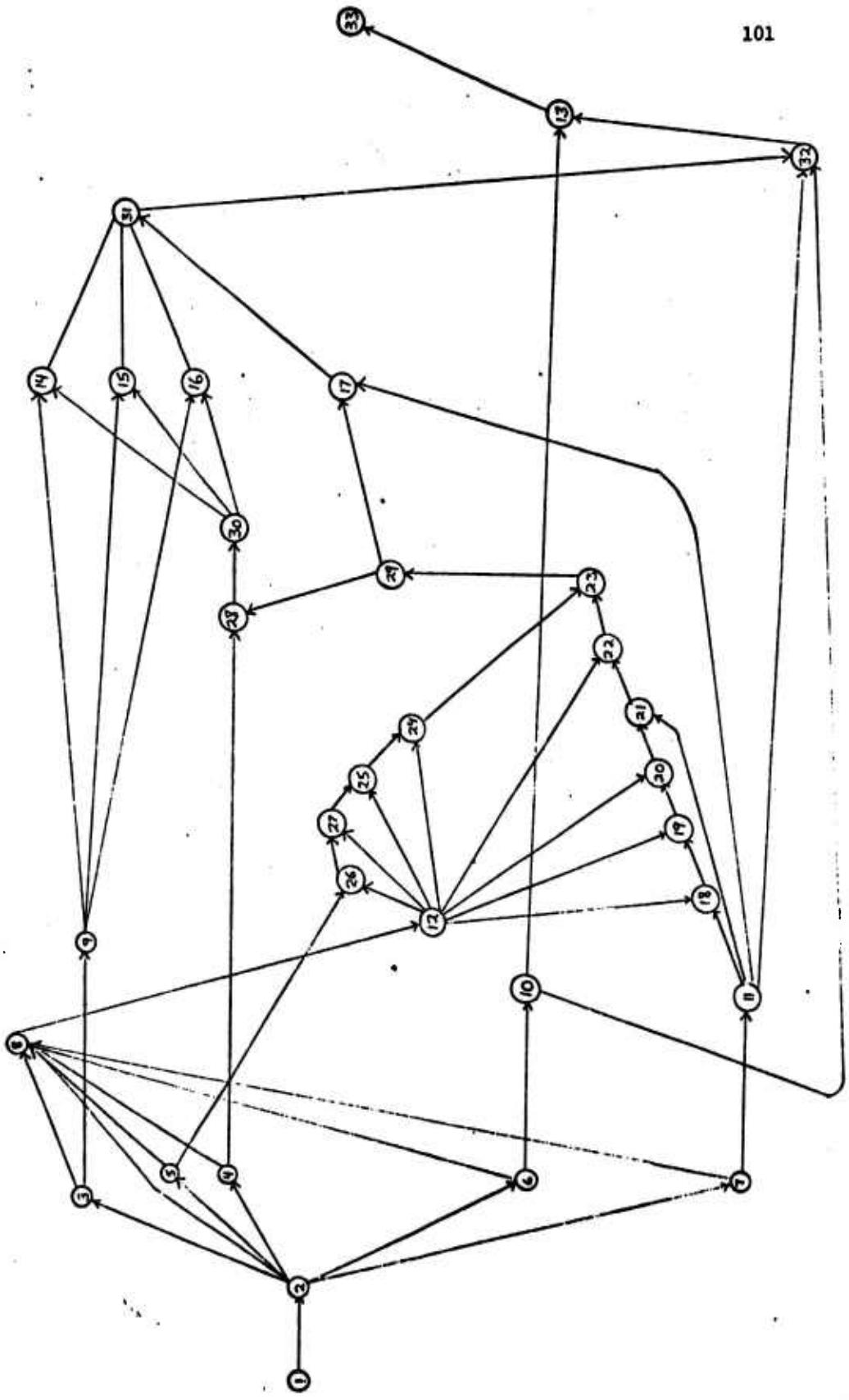
- Col. 11-15: The origin node of the activity, Format (I5)
Col. 21-25: The terminal node of the activity, Format (I5)
Col. 31-40: Parameter 1. The 15-th percentile of the activity's completion time distribution, Format (F10.4).
Col. 41-50: Parameter 2. The 85-th percentile of the activity's completion time distribution, Format (F10.4)

The nodes should be numbered 1, 2, ..., n with the sink being number n.

Example:

The program's input and output are illustrated in terms of the network in Figure D-1.

Figure D-1. Monte Carlo PERT Simulation Program Example Network



58 33

01	5	2		
02	1	2		
03	8	12	336.27	429.47
04	2	3	57.47	89.96
05	2	4	57.47	89.96
06	2	5	57.47	89.96
07	2	6	57.47	89.96
08	3	7	57.47	89.96
09	4	8	68.46	107.95
10	5	8	68.46	107.95
11	6	8	68.46	107.95
12	7	8	68.46	107.95
13	2	8	150.36	193.49
14	6	10	333.96	403.85
15	3	9	333.96	403.85
16	7	11	355.10	409.85
17	11	18	141.75	221.90
18	10	13	672.36	783.11
19	9	14	560.89	660.00
20	9	15	560.89	660.00
21	9	16	560.89	660.00
22	11	17	542.80	638.71
23	12	18	111.10	173.92
24	18	19	256.03	346.98
25	12	19	302.80	400.67
26	12	20	311.95	410.71
27	11	21	423.58	530.74
28	12	22	315.35	415.71
29	19	20	7.66	11.99
30	20	21	16.77	22.55
31	21	22	11.49	17.99
32	22	23	39.54	48.87
33	12	26	301.91	400.86
34	5	26	767.09	892.74
35	12	27	350.31	460.06
36	12	24	382.32	464.98
37	12	25	385.28	461.54
38	25	24	11.49	17.99
39	24	23	16.28	23.00
40	26	27	7.66	11.99
41	27	25	20.86	28.29
42	4	28	810.17	976.10
43	23	29	15.32	23.99
44	28	30	15.32	23.99
45	14	31	57.47	89.96
46	16	31	49.81	77.96
47	15	31	53.64	83.96
48	17	31	88.12	137.94
49	31	32	3.83	6.00
50	32	13	49.81	77.96
51	13	33	109.01	152.63
52	11	32	745.50	811.32
53	10	32	714.74	799.83
54	30	14		
55	30	16		
56	30	15		
57	29	17		

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

58

29

28

103

SAMPLE OUTPUT

	INITIAL INPUT	ACTIVITY	ORIGIN	TERMINAL	PARAMETER 1	PARAMETER 2	PARAMETER 3 (NONE SPECIFIED)	PARAMETER 4 (NONE SPECIFIED)	PARAMETER 5 (NONE SPECIFIED)
1	1	2	2	0	0.0	0.0			
2	2	8	2	12	336.2700	429.4700			
3	3	2	3	2	57.4700	89.7600			
4	4	2	4	5	57.4700	89.7600			
5	5	2	5	6	57.4700	89.7600			
6	6	2	6	7	57.4700	89.7600			
7	7	2	7	8	68.9600	89.9600			
8	8	3	8	8	68.9600	107.9500			
9	9	4	8	8	68.9600	107.9500			
10	10	5	8	8	68.9600	107.9500			
11	11	6	8	8	68.9600	107.9500			
12	12	7	8	8	68.9600	107.9500			
13	13	2	5	10	150.3800	193.4400			
14	14	6	10	10	333.9600	403.8500			
15	15	3	9	11	333.9600	403.8500			
16	16	7	11	11	355.1000	409.8500			
17	17	11	18	18	161.7500	221.9000			
18	18	10	13	13	672.3603	783.1100			
19	19	9	14	14	560.8900	660.0000			
20	20	9	15	15	560.8900	660.0000			
21	21	9	16	16	560.8900	660.0000			
22	22	11	17	17	562.8000	638.7100			
23	23	12	18	18	111.1000	173.9200			
24	24	18	19	19	256.0300	346.9800			
25	25	12	19	20	302.8000	400.6700			
26	26	12	20	20	311.9500	410.7100			
27	27	11	21	21	423.5800	530.7400			
28	28	12	22	22	315.3500	415.7100			
29	29	19	20	20	7.6600	11.9900			
30	30	20	21	21	16.7700	22.5500			
31	31	21	22	22	11.4900	17.9400			
32	32	22	23	23	39.5400	48.8700			
33	33	12	24	24	301.9100	400.8600			
34	34	5	26	26	767.0900	892.7400			
35	35	12	27	27	350.3100	460.0600			
36	36	12	26	26	382.2200	464.9800			
37	37	12	25	25	383.2800	461.5400			
38	38	25	24	24	11.4900	17.9400			
39	39	24	23	23	16.2800	23.0000			
40	40	26	27	27	7.6600	11.9900			
41	41	27	27	27	20.8600	28.2900			
42	42	4	28	28	810.1700	976.1000			
43	43	23	29	29	15.3200	23.9900			
44	44	28	30	30	15.3200	23.9900			
45	45	14	31	31	57.4700	89.9600			
46	46	14	31	31	49.8100	77.9400			
47	47	15	31	31	53.6400	83.9600			
48	48	17	32	32	88.1200	137.9400			
49	49	31	32	32	3.3300	6.0000			
50	50	32	13	13	49.8100	77.9400			
51	51	13	33	33	109.0100	152.6300			
52	52	11	32	32	745.5000	811.3200			
53	53	10	32	32	716.7400	799.8300			
54	54	30	14	14	0.0	0.0			
55	55	30	16	16	0.0	0.0			
56	56	30	15	15	0.0	0.0			
57	57	29	17	17	0.0	0.0			

COPY AVAILABLE TO DOC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

58 29 28 0.0 0.3

THE CRITICAL PATH TIMES FOR 5 TRIALS ARE AS FOLLOWS: (TIME/OBSERVED PERCENTILE)

0.148370 ⁺⁰⁴ 0.203000 ⁺⁰⁰	0.152120 ⁺⁰⁴ 0.400000 ⁺⁰⁰	0.153220 ⁺⁰⁴ 0.600000 ⁺⁰⁰	0.153270 ⁺⁰⁴ 0.800000 ⁺⁰⁰	0.155570 ⁺⁰⁴ 0.100000 ⁺⁰¹
--	--	--	--	--

```

C          1
C          2
C          3
C          4
C          5
C          6
C          7
C          8
C          9
C          10
C          11
C          12
C          13
C          14
C          15
C          16
C          17
C          18
C          19
C          20
C          21
C          22
C          23
C          24
C          25
C          26
C          27
C          28
C          29
C          30
C          31
C          32
C          33
C          34
C          35
C          36
C          37
C          38
C          39
C          40
C          41
C          42
C          43
C          44
C          45
C          46
C          47
C          48
C          49
C          50
C          51
C          52
C          53
C          54
C          55
C          56
C          57
C          58
C          59

C          MONTE CARLO PERT

C          IMPLICIT REAL*8 (A-H,O-Z)
C          FOR THE SAKE OF IDENTIFYING THE APPROPRIATE DIMENSIONS, LET
C          M = THE NUMBER OF ACTIVITIES IN THE NETWORK
C          NMM = NUMBER OF NODES IN THE NETWORK
C          NMMP1 = NMM + 1
C          N = M + NMM
C          NTRIAL = NUMBER OF SIMULATED SETS OF ACTIVITY
C          COMPLETION TIMES
C          NPARM = NUMBER OF PARAMETERS NEEDED FOR GENERATING RANDOM
C          ACTIVITY COMPLETION TIMES FROM A PARTICULAR DISTRIBUTION

C          INTEGER TAIL( M ),HEAD( M ),PARM(M,NPARM),TIMES(NTRIAL)
C          DIMENSION B1INV(NMMP1,NMMP1),CTIME(N),XBI(NMMP1),Y1(NMMP1)
C          DIMENSION REDCOS(N),ISTAT(N),INBASE(NMM),CDF(5),NSPEC(20)

C          CURRENTLY THE DIMENSIONS ARE SET FOR
C          M=60
C          NMM=40
C          NTRIAL=2000

COMMON B1INV,REDCOS,CTIME,XBI,INBASF,HEAD,TAIL,NMMP1,NMM,N,ISTAT
COMMON M,MP1
INTEGER TAIL(60),HEAD(60),TRIAL,NSPEC(20)
DIMENSION INBASE(40),PARM(60,5),CDF(5),TIMFS(2000)
DIMENSION B1INV(41,41),CTIME(100)
DIMENSION XBI(41),Y1(41),REDCOS(100),ISTAT(100)
DATA NSPEC(1),NSPEC(5),NSPEC(9),NSPEC(13),NSPEC(17)/5*'NON'/
DATA NSPEC(2),NSPEC(6),NSPEC(10),NSPEC(14),NSPEC(18)/5*'SP'/
DATA NSPEC(3),NSPEC(7),NSPEC(11),NSPEC(15),NSPEC(19)/5*'ECIF'/
DATA NSPEC(4),NSPEC(8),NSPEC(12),NSPEC(16),NSPEC(20)/5*'IED'/
DATA BLANKS/'      '
M = THE NUMBER OF ACTIVITIES IN THE NETWORK
NMM = THE NUMBER OF NODES IN THE PERT NETWORK
READ(5,100) M,NMM
100 FORMAT(2I3)
N=NMM+M
MP1=M+1
NMMP1=NMM+1
TRIAL = C

C          SIMULATION VARIABLES

C          NTRIAL = NUMBER OF SIMULATED SETS OF ACTIVITY
C          COMPLETION TIMES
C          NPARM = NUMBER OF PARAMETERS NEEDED FOR GENERATING RANDOM
C          ACTIVITY COMPLETION TIMES FROM A PARTICULAR DISTRIBUTION
C          TIMES = VECTOR CONTAINING THE OPTIMUM VALUE FOR EACH TRIAL
C          PARM(I,1) = THE LOWER PERCENTILE POINT FOR THE I-TH ACTIVITY
C          PARM(I,2) = THE UPPER PERCENTILE POINT FOR THE I-TH ACTIVITY

C          READ (5,2501) NTRIAL,NPARM

C          THE ACTIVITIES ARE DESCRIBED IN TERMS OF THEIR NODES

```

```

C      II=THE TAIL NODE, THE ORIGIN NODE          60
C      JJ=THE HEAD NODE, THE TERMINAL NODE        61
C
C      DO F10 I=1,M                             62
C      READ(5,2501) II,JJ,(PARM(I,J),J=1,NPARM) 63
C      2501 FORMAT(10X,I5,5X,I5, 5X,5F10.4)       64
C      TAIL(I) = II                            65
C      610 HEAD(I)=JJ                           66
C      NNPARM = 4*NPARM                         67
C      DO P22 I=1,NNPARM                        68
C      222 NSPEC(I) = BLANKS                     69
C      WRITE(6,2700)                            70
C      2700 FORMAT(1H1,15X,'INITIAL INPUT')       71
C      WRITE(6,2701)                            72
C      2701 FORMAT(1H0,10X,'ACTIVITY ORIGIN TERMINAL PARAMETER 1      P
C      *PARAMETER 2      PARAMETER 3      PARAMETER 4      PARAMETER 5') 73
C      WRITE(6,2509) (NSPEC(I),I=1,20)           74
C      2509 FORMAT (41X,4A4,4(1X,4A4))            75
C      DO 2704 I=1,M                            76
C      2704 WRITE(6,2702) I,TAIL(I),HEAD(I),(PARM(I,J),J=1,NPARM) 77
C      2702 FORMAT(1H ,13X,13.5X,13.7X,13.1X,5(7X,F10.4))        78
C
C      DO 104 I=1,NMM                          79
C      104 INBASE(I)=M+I                         80
C      DO 2001 J=1,M                            81
C      2001 ISTAT(J)=0.                          82
C      DO 2002 J=MP1,N                          83
C      2002 ISTAT(J)=1.                          84
C      DO 10 II=1,NMMP1                         85
C      DO 12 L=1,NMMP1                         86
C      12 B1INV(L,II) = 0.                      87
C      10 B1INV(II,II) = 1.                      88
C      DO 30 I=1,NMM                          89
C      30 XB1(I) = 0.                          90
C      XB1(NMMP1) = 1.                         91
C      TOLR1=1.0D-10                         92
C
C      DO 55610 I=MP1,N                         93
C      55610 CTIME(I) = 0.                       94
C      350 CONTINUE                           95
C
C      GENERATE A SET OF ACTIVITY COMPLETION TIMES          96
C
C      TRIAL = TRIAL + 1                         97
C      CALL PANTIM (CTIME,PARM,M)                98
C      CALL BINVA (82800)                         99
C
C      START THE SIMPLEX ALGORITHM              100
C      SOLVE THE DUAL PROBLEM                  101
C      THE NUMBER OF VARIABLES IS M REAL + NMM SLACKS 102
C      FOR A TOTAL OF N VARIABLES             103
C
C      2800 DO 23 J=1,N                         104
C      RATS = 0.                                105
C      IF (ISTAT(J).EQ.1) GO TO 52800          106
C      IF (J.GT.M) GO TO 22                      107
C      RATS = -B1INV(1,HEAD(J)+1)+B1INV(1,TAIL(J)+1) + CTIME(J) 108
C      GO TO 52800                            109
C      22 RATS = -B1INV(1,J-M+1)                 110
C      52800 REDCOS(J)= RATS                   111
C      23 CONTINUE                           112

```

```

22800 CONTINUE          120
  IRMAX=1               121
  PMAX=PFDCOS(1)        122
  DO 24 J=2,N            123
  IF(PFDCOS(J) .LE. RMAX) GO TO 24  124
  RMAX=REDCOS(J)         125
  IRMAX=J               126
24   CONTINUE             127
  IF(RMAX .LE. TOLR1) GO TO 401  128
22824 CONTINUE             129
  DO 26 L=1,NMMP1        130
  IF (IRMAX.GT.M) GO TO 50026  131
  Y1(L) =-B1INV(L,TAIL(IRMAX)+1)+B1INV(L,HEAD(IRMAX)+1)  132
  GO TO 26               133
50026 Y1(L) = B1INV(L,IRMAX-M+1)  134
26   CONTINUE             135
  Y1(1) = Y1(1) - CTIME(IRMAX)  136
  NUMBER=0               137
  DO 27 L=2,NMMP1        138
27   IF(Y1(L) .LE. TOLR1) NUMBER=NUMBER+1  139
  IF(NUMBER .EQ. NMM) GO TO 403  140
  RMIN=.99D 20            141
  IRMIN=0.                142
  DO 32 II=2,NMMP1        143
  IF(Y1(II).LE. TOLR1) GO TO 32  144
  RATS =XB1(II)/Y1(II)      145
  RR=RATS-RMIN            146
  IF(RR .GE. 0.D0) GO TO 32  147
  RMIN=RATS               148
  IRMIN=II                149
32   CONTINUE             150
  DO 33 J=2,NMMP1        151
  WW=B1INV(IRMIN ,J)/Y1(IRMIN )  152
  DO 37 L=1,NMMP1        153
37   B1INV(L,J)=B1INV(L,J)-WW*Y1(L)  154
33   B1INV(IRMIN ,J)=WW           155
C
C       UPDATE THE BASIC VARIABLES: INBASE AND XB1
C
  ISTAT(INBASE(IRMIN-1))=C  156
  ISTAT(IRMAX)=1            157
  INBASE(IRMIN-1)=IRMAX     158
  W=XB1(IRMIN )/Y1(IRMIN )  159
  DO 38 I=1,NMMP1          160
38   XB1(I)=XB1(I)-Y1(I)*W  161
  XB1(IRMIN )=W            162
  GO TO 2800               163
403  WRITE(6,530)
530  FORMAT(1H0,5X,'NO FEASIBLE SOLUTION EXISTS. CHECK YOUR INPUT DATA
*.*')
  WRITE(6,850)
850  FORMAT(1H1)
  GO TO 949               164
C
C       END OF THE SIMPLEX ALGORITHM
C
401  TIMES(TRIAL) = B1INV(1,NMMP1)  165
  IF (TRIAL.LT.NTRIAL) GO TO 350  166
C
C       INBASE IS A SET OF NMM INTEGER VARIARLES WHICH INDICATE THE

```

C COMPOSITION OF THE CURRENT BASIS. FOR EXAMPLE,
C INBASE(K) = 7 IMPLIES THAT THE K-TH COLUMN IN THE BASIS B
C CORRESPONDS TO THE 7-TH VARIABLE
C
C ISTAT INDICATES THE BASIC STATUS OF EACH VARIABLE
C ISTAT(K) = 1 IMPLIES THAT THE K-TH VARIABLE IS IN THE
C DUAL BASIS
C ISTAT(K) = 0 IMPLIES THAT THE K-TH VARIABLE IS NOT IN THE
C DUAL BASIS
C
C ORDER THE RANDOMLY GENERATED NETWORK COMPLETION TIMES
C
3000 LIMIT = NTRIAL
IPASS = NTRIAL-1
ICHNG = 1
DO 4001 J=1,IPASS
LIMIT = LIMIT-1
IF (ICHNG,FQ,0) GO TO 3060
ICHNG = 0
DO 4002 I=1,LIMIT
IF (TIMES(I),LF,TIMES(I+1)) GO TO 4002
TEMP = TIMES(I+1)
TIMES(I+1) = TIMES(I)
TIMES(I) = TEMP
ICHNG = 1
4002 CONTINUE
4001 CONTINUE
C
C DESCRIBE THE ORDERED NETWORK COMPLETION TIMES
C
3060 WRITE(6,9662)
9662 FORMAT(1H1)
WRITE(6,3012) NTRIAL
3012 FORMAT (1H0,10X,'THE CRITICAL PATH TIMES FOR ',I5,' TRIALS ARE AS
* FOLLOWS: (TIME/OBSERVED PERCENTILE)')
LINE = (NTRIAL+4)/5
DO 3050 J=1,LINE
I2 = (J-1)*5+1
I3 = J*5
ICNT = 0
IF (NTRIAL-I2+1).LT.5) I3= NTRIAL
DO 3051 K=I2,I3
ICNT = ICNT+1
3051 CDF(ICNT) = DFLOAT(K)/DFLOAT(NTRIAL)
WRITE (6,3011) (TIMES(I),I= I2,I3)
3011 FORMAT(1H0,5X,5(6X,F15.5))
WRITE (6,3013) (CDF(K),K=1,ICNT)
3013 FORMAT (6X,5(6X,F15.5))
3050 CONTINUE
. 999 STOP
END
SUBROUTINE BINVA(*)
IMPLICIT RFAL*8 (A-H,O-Z)
COMMON BIINV,REDCOS,CTIME,XB1,INBASE,IHEAD,ITAIL,NMNP1,NMM,N,ISTAT
COMMON M,MPI
DIMENSION ISTAT(100),IHEAD(60),ITAIL(60),XB1(41)

180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239

DIMENSION B1INV(41,41),INBASE(40),CTIME(100),REDCOS(100)

240

C UPDATE THE FIRST ROW OF B1INV AFTER CHANGING CTIME

241

C DO 1 I=2,NMMP1

242

R B1INV(1,I) = 0.0

243

DO 1 J=2,NMMP1

244

I B1INV(1,I) = B1INV(1,I) + B1INV(J,I)*CTIME(INBASE(J-1))

245

C UPDATE VALUE OF THE OBJECTIVE FUNCTION

246

C XB1(I) = B1INV(1,NMMP1)

247

RRETURN1

248

END

249

SUBROUTINE RANTIM (CTIMF,PARM,M)

250

C SUBROUTINE RANTIM GENERATES M RANDOM TIMES FROM A SPECIFIED
DISTRIBUTION WITH PARAMETERS CONTAINED IN PARM(60,5) AND RETURNS
WITH THE RESULTS IN CTIMF(99).

251

252

253

254

C IMPLICIT REAL*8 (A-H,O-Z)

255

C DATA J/0/,IY/19447/,TPI/6.2831853/

256

C DIMENSION PARM(60,5),CTIME(99),SAVTIM(99)

257

C IF (J.NE.0) GO TO 30

258

C THE FOLLOWING GENERATES A CHI SQUARE RANDOM DEVIATE WITH 3 DF'S
TRANSFORMED TO MAKE THE LOWER POINT CORRESPOND TO THE 15-TH
PERCENTILE AND THE UPPER POINT CORRESPOND TO THE 85-TH PERC.

259

C PARM(3,I) = THE PERCENTILE DIFFERENCE

260

C DO 5 I=1,M

261

S PARM(I,3) = PARM(I,2)-PARM(I,1)

262

5 IF (PARM(I,3).EQ.0) CTIME(I) = PARM(I,1)

263

30 J = J+1

264

IF (MOD(J,2).EQ.1) GO TO 20

265

C DO 15 I=1,M

266

15 IF (PARM(I,3).EQ.0) GO TO 15

267

CTIME(I) = SAVTIM(I)

268

CONTINUE

269

RRETURN

270

C GENERATE A PAIR OF COMPLETION TIMES FOR EACH ACTIVITY.

271

C THE BOX-MULLER METHOD IS USED TO GENERATE A PAIR OF

272

C NORMAL DEVIATES

273

C U1 AND U2 = UNIFORM RANDOM NUMBERS

274

C W*DJSIN(AN) = STANDARD NORMAL RANDOM VARIABLE

275

C W*DCOS(AN) = STANDARD NORMAL RANDOM VARIABLE

276

C CHI3 = A CHI-SQUARE RANDOM VARIABLE -- GENERATED USING THE

277

C METHOD OF WILSON AND HILFFRTY - PROC. NAT. ACADEMY OF

278

C SCIENCE, 1931

279

C1 AND C2 TRANSFORM THE CHI-SQUARE WITH 3 D.F.

280

20 DO 3002 I=1,M

291

3002 IF (PARM(I,3).EQ.0) GO TO 3002

292

IY = IY*65539

293

IF (IY) 3015,3016,3016

294

3015 IY = IY + 2147483647 + 1

295

296

297

298

299

```

3016 YFL = IY          300
    U1 = YFL*.465613E-9 301
    IY = IY*65539        302
    IF (IY) 3025,3026,3026 303
3025 IY = IY + 21474E3647 + 1 304
3026 YFL= IY          305
    U2 = YFL*.465613E-9 306
    W = -2.*DLNG(U2)    307
    W = DSQRT(W)        308
    AN = TPI*U1          309
    C2 = 4.51927/PARM(1,3) 310
    C1 = .79777-PARM(1,1)*C2 311
    CTIMF(I) = ((.392530614*W*DSIN(AN)+1.335416269)**3-C1)/C2 312
    SAVTIM(I) = ((.392530614*W*DCOS(AN)+1.335416269)**3-C1)/C2 313
3002 CONTINUE          314
    RRETURN             315
    END                 316

```

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Texas A&M University		2a. REPORT SECURITY CLASSIFICATION Unclassified
		2b. GROUP Unclassified
3. REPORT TITLE STATISTICAL PERT: AN IMPROVED SUBNETWORK ANALYSIS PROCEDURE.		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Progress Report		
5. AUTHOR(S) (First name, middle initial, last name) R. L. Sielken, Jr., H. O. Hartley, R. K. Spoerl		
6. REPORT DATE 11 NOV 1975		
7b. TOTAL NO. OF PAGES 111		
7d. NO. OF REFS 1		
8c. CONTRACT OR GRANT NO. 15 N00014-68-A-0140		
8e. PROJECT NO. NRO 47-700		
8f. NR - 47-700		
9b. ORIGINATOR'S REPORT NUMBER(S) Number 51 14 THEMIS-TR-51		
9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)		
10. DISTRIBUTION STATEMENT This document has been approved for public release and sale; its distribution is unlimited		
11. SUPPLEMENTARY NOTES See page following Table of Contents		12. SPONSORING MILITARY ACTIVITY Office of Naval Research
13. ABSTRACT 9 THEMIS optimization research program		

347 380

ATTACHMENT III

Unclassified

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Statistical PERT						
Subnetwork Analysis						
Project Scheduling						
Two Point Approximation						
Stochastic Networks						

**BASIC DISTRIBUTION LIST FOR
UNCLASSIFIED TECHNICAL REPORTS**

Operations Research Office of Naval Research (Code 434) Arlington, Virginia 22217	Copies	Director Office of Naval Research Branch Office 495 Summer Street Boston, Massachusetts 02210 Attn: Dr. A.L. Powell	Copies
Director, Naval Research Laboratory Attn: Library, Code 2029 (ONRL) Washington, D.C. 20390	3	Director Office of Naval Research Branch Office 219 South Dearborn Street Chicago, Illinois 60604 Attn: Dr. A.R. Dawe	1
Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2	Office of Naval Research San Francisco Area Office 760 Market St. - Room 447 San Francisco, California 94103	1
Defense Logistics Studies Information Exchange Army Logistics Management Center Fort Lee, Virginia 23801	12	Technical Library Naval Ordnance Station Indian Head, Maryland 20640	1
Technical Information Center Naval Research Laboratory Washington, D.C. 20390	1	Bureau of Naval Personnel Navy Department Technical Library Washington, D.C. 20370 STOP 82	2
Office of Naval Research New York Area Office 207 West 24th Street New York, New York 10011 Attn: Dr. J. Laderman	6	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	1
Director, Office of Naval Research Branch Office 1030 East Green Street Pasadena, California 91101 Attn: Dr. A.R. Laufer	1	Naval Ordnance Station Louisville, Kentucky 40214	1
Lt. Dennis Johnston - ED3 Manned Spacecraft Center NASA Houston, Texas 77058	1	Library Naval Electronics Laboratory Center San Diego, California 92152	1
Mr. William Dejka Code 360 Naval Electronics Laboratory Center San Diego, California 92132	1	Naval Ship Engineering Center Philadelphia Division Technical Library Philadelphia, Pennsylvania 19112	1
Professor Lucien A. Schmit, Jr. Room 6731, Boelter Hall School of Engrng. & Applied Sci. University of California Los Angeles, California 90024	1	Dr. A.L. Slafkosky Scientific Advisor Commandant Marine Corps (Code AX) Washington, D.C. 20380	1

Purdue University Graduate School of Ind. Admin. Lafayette, Indiana 47907 Attn: Prof. Andrew Whinston	Copies 1	Dr. Del Cilmer Naval Weapons Center (Code 607) China Lake, California 93555	Copies 1
Department of Economics Tisch Hall - 5th Floor Washington Square, NY Univ. New York City, NY 10003 Attn: Prof. O. Morgenstern	1	Dr. Paul Murrill Project Themis Department of Chemical Engineering Louisiana State University Baton Rouge, Louisiana 70803	1
Dept. of Statistics Syntex Research 3401 Hillview Palo Alto, CA 94304 Attn: Prof. Stuart Bessler	1	Director Army Materials & Mechanics Research Center Attn: Mr. J. Bluhm Watertown, Massachusetts 02172	1
University of California Department of Engineering Los Angeles, California 90024 Attn: Prof. R.R. O'Neill	1	Commanding Officer U.S. Army Ballistic Research Laboratories Attn: Dr. John H. Giese Aberdeen Proving Ground, Maryland 21005	1
Maritime Transportation Research Board National Academy of Sciences 2101 Constitution Avenue Washington, D.C. 20418 Attn: RADM J.B. Oren USCG (RET)	1	University of Iowa Department of Mechanics and Hydraulics Iowa City, Iowa 52240 Attn: Prof. W.F. Ames	1
Stanford University Department of Operations Research Stanford, California 94305 Attn: Prof. A.F. Veinott, Jr.	1	Office of Naval Research Resident Representative 1105 Guadalupe Lowich Building Austin, Texas 78701 Attn: Mr. Francis M. Lucas	1
Texas A&M Foundation College Station, Texas 77843 Attn: Prof. H.O. Hartley	1	Dr. Jerome Bracken Institute of Defense Analyses 400 Army-Navy Drive Arlington, Virginia 22202	1
Case Western Reserve University Cleveland, Ohio 44106 Attn: Prof. B.V. Dean	1	Professor Richard L. Fox School of Engineering Case Western Reserve University Cleveland, Ohio 44106	1
University of California Center for Research in Management Science Berkeley, California 94720 Attn: Prof. W.I. Zangwill	1	Director, National Security Agency Ft. George G. Meade, Maryland 20755 Attn: Dr. Joseph Blum, R44	1
U.S. Naval Postgraduate School Department of Operations Research and Economics Monterey, California 93940 Attn: Prof. C.R. Jones			

Copies	Harvard University Department of Statistics Cambridge, Massachusetts 02139 Attn: Prof. W.G. Cochran	Copies
1	Naval Applied Science Laboratory Technical Library, Code 222 Flushing & Washington Avenues Brooklyn, New York 11251	1
1	Naval Undersea Warfare Center 3202 E. Foothill Boulevard Pasadena, California 91107 Technical Library	1
1	University of Chicago Statistical Research Center Chicago, Illinois 60637 Attn: Prof. W. Kruskal	1
1	Stanford University Department of Statistics Stanford, California 94305 Attn: Prof. G.J. Lieberman	1
1	Florida State University Department of Statistics Tallahassee, Florida 32306 Attn: Prof. I.R. Savage	1
1	Princeton University Department of Mathematics Princeton, New Jersey 08540 Attn: Prof. J.W. Tukey	1
1	Stanford University Department of Statistics Stanford, California 94305 Attn: Prof. T.W. Anderson	1
1	University of California Department of Statistics Berkeley, California 94720 Attn: Prof. P.J. Bickel	1
1	University of Washington Department of Mathematics Seattle, Washington 98105 Attn: Prof. Z.W. Birnbaum	1
1	Cornell University Department of Mathematics White Hall Ithaca, New York 14850 Attn: Prof. J. Kiefer	1
1	Columbia University Department of Industrial Engineering New York, New York 10027 Attn: Prof. C. Derman	1
1	New York University Institute of Mathematical Sciences New York, New York 10453 Attn: Prof. W.M. Hirsch	1
1	University of North Carolina Statistics Department Chapel Hill, North Carolina 27515 Attn: Prof. W.L. Smith and Prof. M.R. Leadbetter	1
1	Purdue University Division of Mathematical Sciences Lafayette, Indiana 47079 Attn: Prof. H. Rubin	1
1	University of California, San Diego Department of Mathematics P.O. Box 109 La Jolla, California 92038 Attn: Prof. M. Rosenblatt	1
1	Florida State University Department of Statistics Tallahassee, Florida 32306 Attn: Prof. R.A. Bradley	1
1	New York University Department of Industrial Engineering & Operations Research Bronx, New York 10453 Attn: Prof. J.H.K. Kao	1
1	University of Wisconsin Department of Statistics Madison, Wisconsin 53706 Attn: Prof. G.E.P. Box	1

Logistics Research Project	Copies	The University of Michigan	Copies
The George Washington University		Department of Mathematics, W.E.	
707 - 22nd Street, N.W.		Ann Arbor, Michigan 48104	
Washington, D. C. 20037		Attn: Prof. R.M. Thrall	1
Attn: Dr. W. H. Marlow	1		
International Business Machines Corporation		Princeton University	
P.O. Box 218, Lamb Estate		Department of Mathematics	
Yorktown Heights, New York 10598		Princeton, New Jersey 08540	
Attn: Dr. Alan Hoffman	1	Attn: Prof. A.W. Tucker	1
University of California Management Sciences Research Project		Case Western Reserve University	
Los Angeles, California 90024		Systems Research Center	
Attn: Dr. J.R. Jackson	1	Cleveland, Ohio 44106	
		Attn: Prof. M. Mesarovic	1
Harvard University		University of Texas	
Department of Economics		Department of Mathematics	
Cambridge, Massachusetts 02138		Austin, Texas 78712	
Attn: Prof. K.J. Arrow	1	Attn: Dr. A. Charnes	1
Cowles Commission for Research in Economics		Stanford University	
Yale University		Department of Operations Research	
New Haven, Connecticut 06520		Stanford, California 94305	
Attn: Prof. Martin Shubik	1	Attn: Dr. D.L. Iglehart	1
Carnegie-Mellon University		University of Delaware	
Graduate School of Industrial Administration		Department of Mathematics	
Pittsburgh, Pennsylvania 15213		Newark, Delaware 19711	
Attn: Prof. G. Thompson	1	Attn: Prof. R. Kemerage, Jr.	1
University of California		Stanford University	
Department of Economics		Department of Operations Research	
Berkeley, California 94720		Stanford, California 94305	
Attn: Prof. R. Rudner	1	Attn: Prof. F.S. Hillier	1
University of California		Dr. Claude-Alain Burdet	
Operations Research Center		Asst. Prof. Industrial Admin.	
Institute of Engineering Research		Carnegie-Mellon University	
Berkeley, California 94720		Pittsburgh, Pennsylvania 15213	
Attn: Prof. D. Gale	1		
University of California		Stanford University	
Graduate School of Business Administration		Department of Operations Research	
Los Angeles, California 90024		Stanford, California 94305	
Attn: Prof. J. Marschak	1	Attn: Prof. G. B. Dantzig	1
Chief of Naval Research (Code 436)		Science Librarian	
Department of the Navy		Kresge Library	
Arlington, Va. 22217		Oakland University	
		Rochester, Michigan 48063	1

Copies	University of Connecticut Department of Statistics Storrs, Connecticut 06268 Attn: Prof. H.O. Posten	Professor Geoffrey J. Watson, Chairman Department of Statistics Princeton University Fine Hall Princeton, N. J. 08540	Copies
1	Prof. Emanuel Parzen Statistics Department State Univ. New York at Buffalo 4230 Ridge Lea Road Amherst, New York 14226	Stanford University Department of Mathematics Stanford, California 94305 Attn: Prof. S. Karlin	1
1	ARCON Corporation Lakeside Office Park North Avenue at Route 128 Wakefield, Massachusetts 01880 Attn: Dr. A. Albert	University of Sheffield Department of Probability and Statistics Sheffield 10, ENGLAND Attn: Prof. J. Gani	1
1	Stanford University Department of Statistics Stanford, California 94305 Attn: Prof. H. Chernoff	University of California Operations Research Center Institute of Engineering Research Berkeley, California 94720 Attn: Prof. R.E. Barlow	1
1	Yale University Department of Statistics New Haven, Connecticut 06520 Attn: Prof. L.A. Savage	Stanford University Department of Statistics Stanford, California 94305 Attn: Prof. H. Solomon	1
1	Rutgers-The State University Statistics Center New Brunswick, New Jersey 08903 Attn: Prof. H.F. Dodge	Applied Mathematics Laboratory Naval Ships Research Development Center Washington, D.C. 20007	1
1	Yale University Department of Statistics New Haven, Connecticut 06520 Attn: Prof. F. J. Anscombe	Systems Analysis Division Room BE760, Pentagon Washington, D.C. 20350 Attn: Mr. A.S. Rhodes, Op-964	1
1	Purdue University Division of Mathematical Sciences Lafayette, Indiana 47907 Attn: Prof. S.S. Gupta	Department of Statistics University of North Carolina Chapel Hill, North Carolina 27515 Attn: Prof. M.R. Leadbetter	1
1	Cornell University Department of Industrial Engineering Ithaca, New York 19850 Attn: Prof. R.E. Bechhofer	Southern Methodist University Department of Statistics Dallas, Texas 75222 Attn: Prof. D.B. Owen	1
1	Mrs. Barbara Eaudi Univ. Program Coordinator, B.E. NASA Johnson Space Center Houston, TX 77058	Israel Institute of Technology Technion Haifa, ISRAEL Attn: Prof. P. Naor	

Office of Gifts & Exchanges Copies

Library
Texas A&M University
College Station, Texas 77843 2

Archives
Texas A&M University
College Station, TX 77843 1