General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)

CR 86285

GERT SIMULATION PROGRAM II
(GERTS II)

by

A. Alan B. Pritsker and Phillip C. Ishmael

Research Sponsored
by
Electronics Research Center
National Aeronautics and Space Administration
NASA Contract NASA-12-2035

June 1969

FACILITY FORM 602
N70-14197 (ACCESSION NUMBER)
95 (PAGES)
CR-86285 (NASA CR OR TMX OR AD NUMBER)
1 (THRU)
(CODE)
08 (CATEGORY)

JAN 1970
RECEIVED
NASA STI FACILITY
INPUT BRANCH

ABSTRACT

This report describes the procedures for using a digital computer program for simulating GERT networks. New and advanced concepts of GERT are introduced in this report. The simulation program can accommodate GERT networks which have OR and AND logical operations associated with the input side of a node and deterministic or probabilistic operations associated with the output side of a node. A re-definition of a node type permits a generalization of the logical operations that can be associated with a node. Another new characteristic of a node is the number of releases to realize the node after its first realization.

The branches of a GERT network are described in terms of a probability that the branch is realized; a time to perform the activity represented by the branch; a count designation; and an activity number. The time associated with a branch can be a random variable. The count designator identifies a count set for which a counter is indexed every time the branch is realized. The activity number identifies nodes affected by the realization of the branch. Through activity numbers, a network can be modified during the simulation of the network.

The results obtained from the GERT simulation program are:

1. The probability that a node is realized;
2. The average, standard deviation, minimum and maximum time to realize a node;
3. A histogram of the times to realize a node; and
4. The average, standard deviation, minimum, and maximum counts observed during the time to realize a node.

Normally the above information is obtained for each sink node of the network. The program is written to permit the above information to be obtained for any node specified by the input data.

The GERT simulation program is written in FORTRAN IV using the concepts employed in GASP IIA. The program has been exercised on the IBM 1800 and GE 425 computers. The IBM 1800 has a FORTRAN IV operation set more restrictive than the general FORTRAN IV operation set. Therefore, the GERT simulation program can be used on most computers which have FORTRAN IV compilers. GERT networks with up to 1000 nodes can be simulated. Six illustrations of the use of the program are included along with a FORTRAN listing of all subprograms.

TABLE OF CONTENTS

GERT SIMULATION PROGRAM II

INTRODUCTION

Applications of the GERT simulation program reported previously [7,8] resulted in the need for additional capability and concepts and for simplified operating procedures. This need has been satisfied with the completion of GERT Simulation Program II, hereafter referred to as GERTS II.

Beginning with the easy problem of improving program efficiency, the GERT simulation program was stripped of all excess subprograms and statements to reduce computing time and the number of data cards. Many of the features of GASP IIA included in the original GERT simulation program were tailored to meet the specific needs involved in simulating GERT networks. These changes have resulted in a program which can reduce computing time by as much as 50%, reduce the number of data cards by as much as 50% and made available enough storage to increase the number of nodes possible to 1,000. After improving the efficiency of the program, research was performed to increase the capability of the program. The following list describes the new features included in GERTS II.

1. Counters which count the number of times a branch or set of branches is realized prior to the realization of a node.
2. Nodes which are characterized by:
   a. the number of releases required to realize the node for the first time;

b. the number of releases required to realize the node after the first time;

c. the removal of events that are scheduled to release the node; and

d. the method for scheduling the activities emanating from the node (deterministic or probabilistic).

3. Modification of the network based on the occurrence of events during the simulation of the network.

4. A method for tracing a set of simulation runs.

5. Improved printout of the description of the network and the final results.

The main purpose of this report is to describe the procedure for using GERTS II. Since several new concepts associated with GERT have been developed it is necessary to describe these before proceeding with the use of the new program. These are presented in the next section. The program operating procedure is described in general terms in the body of this report. A detailed description of each data card type for GERTS II is presented in Appendix C. The storage limitations and error messages are included in the program operating procedure section of this report. By past experience it has been found that the most convenient way to present a new computer program is by illustrating its operating procedure and use through examples. In presenting GERTS II this approach is followed and six examples called projects 11 through 16 are presented. For each project, the GERT network, data input, function of the network, and the output reports from the simulation program are illustrated. Through the use of these projects it is hoped that both the modeling aspects associated with the new characteristics of GERT and the operating procedure associated with GERTS II will be communicated.

Since GERTS II is written in FORTRAN it is capable of being modified and improved by users. To facilitate the changing of the GERT simulation program to meet specific requirements of a user, a glossary of GERTS II subprograms and variables is presented in Appendix A and a complete FORTRAN listing of GERTS II is presented in Appendix B. In the recommendations section of this report, additional features and improvements are suggested and the means for implementing these recommendations presented.

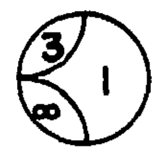
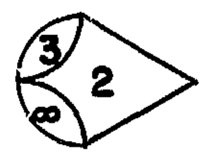
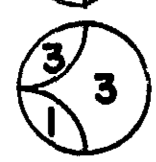
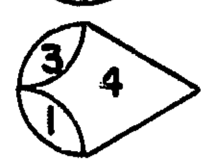
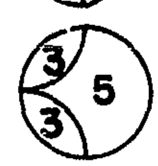
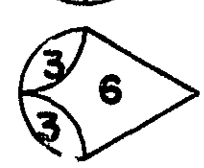
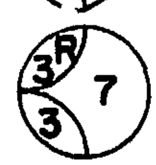
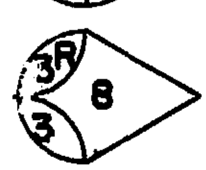
NEW GERT NETWORK CHARACTERISTICS

GERT networks consists of nodes and directed branches. For the GERT simulation program, a node is described in terms of two parameters:

1. The number of releases, i.e., the number of activities leading into the node that must be realized before the node can be realized; and
2. A node type which specifies:
   a. whether the output side of the node is deterministic (all activities emanating from the node are taken) or probabilistic (random sampling is to be performed to determine which one of the activities emanating from the node is taken); and
   b. the number of releases required to have the node realized again.

The number of releases associated with a node specifies the number of times activities incident to the node must be realized before the node can be realized. When the number of releases is 1, the input side of the node can be thought of as an OR operation. If the number of releases equals the number of activities incident to the node, the node can be thought of as an AND operator. However, it is permissible to specify the number of releases to be less than or greater than the number of activities incident on the node. For example, the number of releases can be 2 whereas the number of activities incident on the node could be 3. This would represent the case where if 2 of the 3 activities were realized, the node is realized. Alternatively, the number of releases can be 2 whereas the number of activities incident to the node could be 1. This would represent the case where the activity must be realized twice before the node is realized.

There are 8 node types that can be used in GERTS II. Odd numbered node types, 1, 3, 5 and 7 indicate a deterministic output for the node, whereas even numbered node types represent a probabilistic output side for the node. Node types 1 and 2 cannot be realized more than once during any simulation run of the network. All other node types can be realized as many times as the appropriate conditions exist in the simulation of a network. After the first realization of node types 3 and 4, the realization of any activity incident to the node will cause the node to be realized again. Node types 1 through 4 have not been changed from the original GERT simulation program (7,8). Node types 5 through 8 when realized, have the number of releases associated with the node reset to the initial number of releases assigned to the node. Node types 7 and 8, in addition, cause any end of activity events for activities incident to the node to be removed (cancelled) at the time the node is realized. Note that none of the node types are restricted in terms of the logical operation associated with their input side. Thus GERTS II has generalized the node types associated with GERT. A word of caution is required when using node types 7 and 8. Since end of activity events are removed only at the time the node is realized, it is possible that there exist some activities on forward paths that have not been scheduled that are incident to the node in question. When these activities are scheduled and completed, they will reduce the number of releases associated with the node. Thus, node types 7 and 8 may not correspond to the previous defined INCLUSIVE-OR node.* Figure 1 presents the 8 node types.

Node Type	Scheduled Events Removed When Realized	Reset Number of Releases	Output Type (a)	Example of Symbol (b)
1	no	to	D	
2	no	to	P	
3	no	no	D	
4	no	no	P	
5	no	yes	D	
6	no	yes	P	
7	yes	yes	D	
8	yes	yes	P	

(a) D = Deterministic
P = Probabilistic

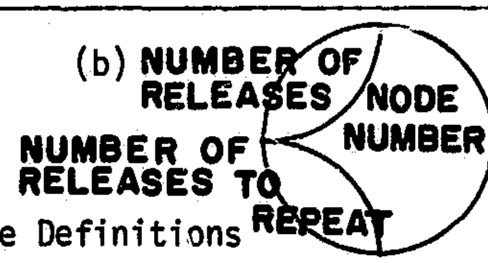
(b) 

Figure 1. Node Type Definitions

*It is possible to accomplish this by using the network modification feature of GERTS II.

The AND and OR operations previously defined related the number of activities incident to a node to the realization of the node. The new node definitions do not involve this relationship. However the AND and OR nodes are easily represented by the new node types, i.e., AND nodes have the number of releases equal to the number of branches incident to the node, and OR nodes have the number of releases equal to 1.

The branches of GERT networks represent activities. Activities emanate from a start node and are incident to an end node. Associated with activities are a probability that the activity will be realized given its start node is realized and a time to perform the activity given the activity is realized. For GERTS II the time variable is specified by a parameter set number and a distribution type. The following six distribution types are available:

1. Constant;
2. Normal;
3. Uniform;
4. Erlang;
5. Lognormal; and
6. Poisson.

The parameters set number along with the distribution type completely describe the time variable associated with an activity. Each distribution type specifies the arrangement of the parameters in a parameter set. This is described further in Appendix C under the description of Data Card Type 8. With GERTS II, two additional characteristics can be associated with an activity. These are a counter type and an activity number.

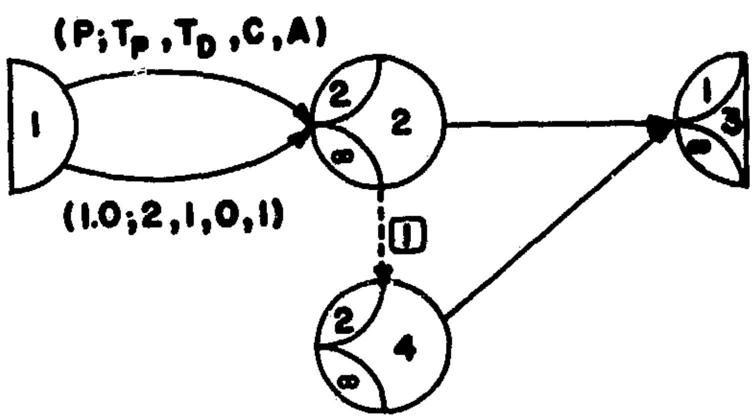
The counter type number specifies the counter to be increased by 1 every time the activity is realized. The number of counter types permitted is limited to 4.* Any number of activities may be associated with a counter type. Statistics are automatically kept on the counter types. At the end of all simulation runs, the average and standard deviation of the number of times a counter type was realized prior to the realization of each node for which statistics are collected is determined and printed. In addition, the minimum and maximum number of times activities having the specified counter type were realized during a simulation is printed. Since the number of counts is always referenced to the realization of a node, the number of counts may be different in different simulation runs due to the times at which the nodes are realized.

Activity numbers are given to activities to permit network modifications based on the realization of the activity. Specification of an activity number does not automatically indicate that the network will be modified. However, only activities with activity numbers can cause the network to be modified. Network modification involves the replacing of a node by another node. The node to be replaced is deleted from the network when and if it is realized. The activities then caused to occur are from the node that is inserted. In this way a node can be changed many times before it is actually realized. Thus, if the realization of activity 1 could cause node 5 to be replaced by node 8, and activity 2 could cause node 5 to be replaced by node 9. Both activities should reference node 5; that is, activity 2 should have node 9 replace node 5, not node 8.

---

*Changes in the dimensions of the arrays KOUNT and SUMA can be made to increase this value.

The activity number causing the network modification along with all the nodes to be replaced, and the nodes to be inserted, are specified by the user on Data Card Type 10. The method for incorporating network modifications is described in the program operating procedure section and Appendix C of this report. Figure 2 illustrates the branch and node modification notation that will be used throughout this report. Modifications will be shown by a dashed branch with the activity number attached in a square. The modification in Figure 2 is read "node 2 is replaced by node 4 when activity 1 is realized."



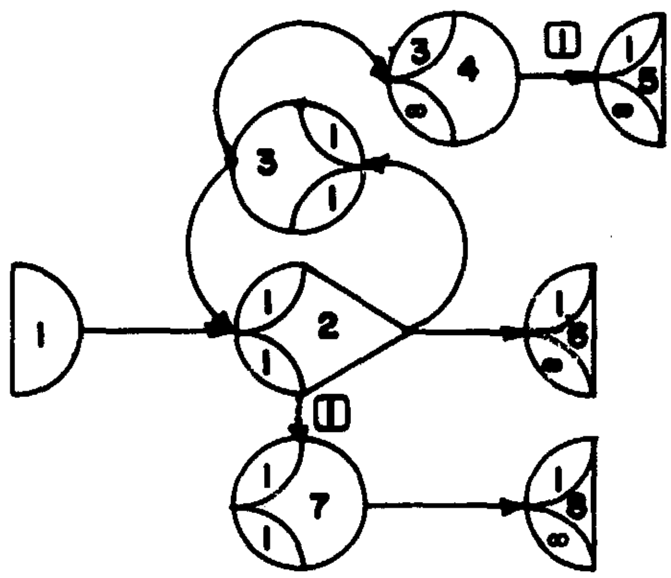
LEGEND

p = probability of realization
t_p = parameter set for time
t_D = distribution type
c = counter type
a = activity number

Node A is replaced by node B when activity with activity number 1 is realized.

Figure 2. Illustration of Branch Descriptors and Network Modification Symbolism.

As an illustration of the power of these new characteristics, the following network is presented.



This network represents the changing of the network structure when the self-loop about node 2 is taken three times and is accomplished in the following manner. The output of node 3 is DETERMINISTIC so that every time node 3 is realized both branches emanating from node 3 are taken. The branch from node 3 to node 4 is used to count the number of times node 3 is realized. Node 4 is only realized when the branch incident to it is realized three times (of course, this corresponds to three traversals of the self-loop). When node 4 is realized, the activity labeled activity number 1 (a 1 on the network in this case) causes node 2 to be replaced by node 7, and the objective of changing the network is achieved.* Of significant importance in the above network is the incor-

---

*Care is required here to ensure that the branch from node 4 to node 5 is realized prior to the realization of the branch from node 3 to node 2. If a zero time is associated with both branches, normal operation would have the branch from node 3 to node 2 realized first since it was scheduled first. By assigning a small negative time (-.000001) to branch from node 4 to node 5, the desired ordering can be obtained.

poration on the network of branches representing activities and branches representing information transfers. The inclusion of different types of branches within a simple network opens the door to the further combining of network types within the GERT framework.

PROGRAM OPERATING PROCEDURE

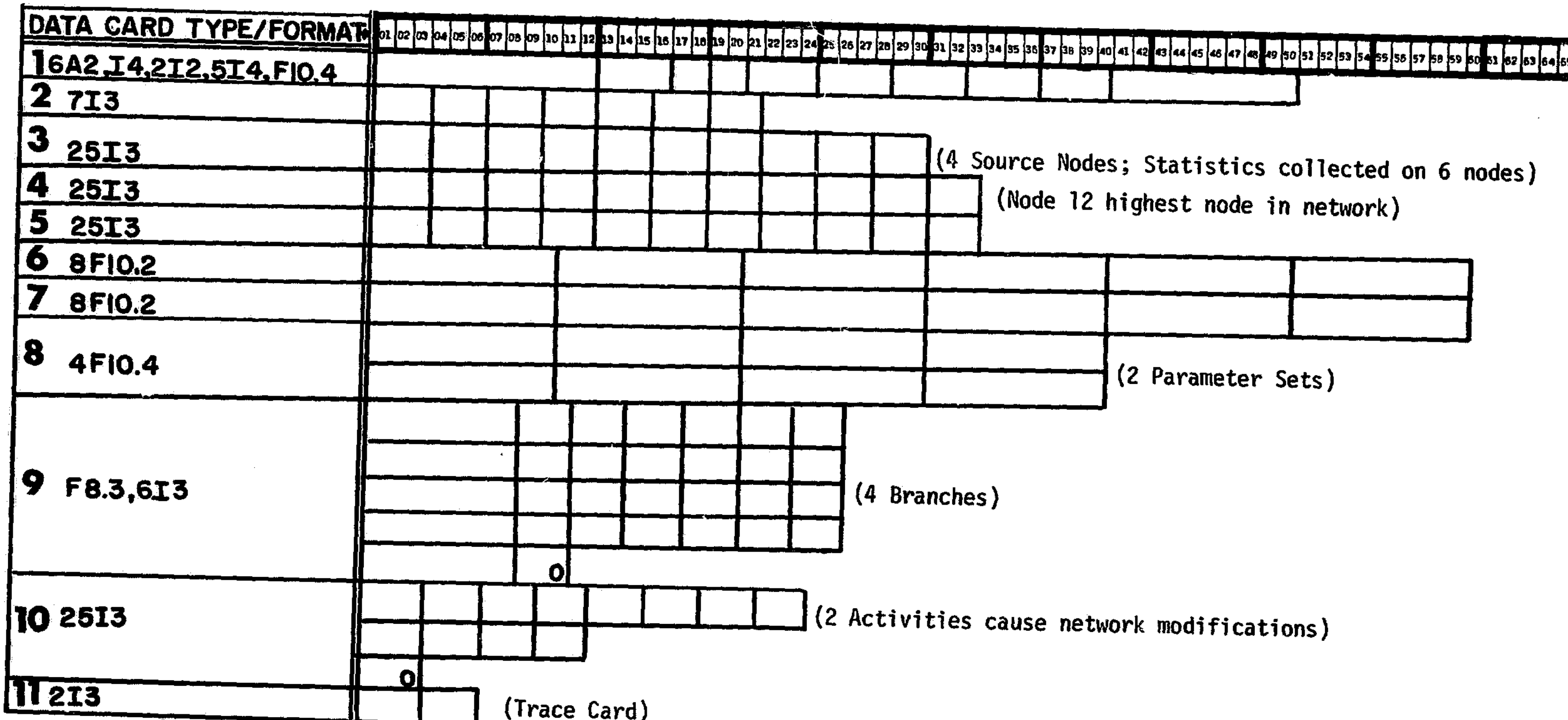
Input Description

The control cards for the program are the standard FORTRAN control cards for the computer being used. A graphical description or data worksheet for the input data is given in Figure 3. In Appendix C, each input data card is described in the order they are read into the computer, and in terms of the fields required. The variable in each field is described and the format and number of variables which must be inserted are given. After card 1, all integer variables are in format I3; probabilities are in format F8.3; and every other real variable requires ten columns.

Limitations on Project Size

Limitations on the size of a network that can be simulated depend on the dimensions of the arrays of GERTS II. By changing array sizes, larger networks can be simulated. The limitations and the arrays causing the limitation are given in tabular form below.

Network Characteristic	Limitation	Arrays Affected
Number of nodes	999	MFE(999),MLE(999),NQ(999), NRELP(999), NREL(999), NTYPE(999), NFTBU(999), NSET(7000), QSET(1000) [Must also change statement GETS 120: IMN = 999]
Number of parameter sets	300	PARAM(300,4)
Number of source nodes	20	NSORC(20)
Number of nodes on which statistics are collected	20	NSINK(20), JSINK(20), XLOW(20), WIDTH(20), JCELS(20,32), SUMA(100,5)
Number of counter types	4	KOUNT(4), SUMA(100,5)



*In terms of requirements for input not the FORTRAN statements

Figure 3. Data Work Sheet for Input to GERT Simulation Program II

Number of nodes affected by an activity	11	NABYA(22) [Must also change statement DATS 1460 in DATAN]
Number of nodes that can be modified	approximately 400	NPO(999), NABA(999)

The dimensions of NSET and QSET are changed in the main program. The dimension of QSET must be larger than the maximum number of nodes plus the maximum number of simultaneous activities. The dimension of NSET is seven times the dimension of QSET. The first dimension of SUMA is the maximum number of counter types plus one times the maximum number of nodes on which statistics are collected. In the current program it is (4+1)*20 = 100. The nodes to be modified are stored in pairs in the array NABA. Each pair consists of the node replaced and the node to be inserted. The nodes affected by an activity are stored sequentially and a zero is used to indicate the end of the nodes affected by a particular activity. For example, the first ten cells of NABA could be

1 6 9 7 8 0 8 3 4 0

which indicates that the realization of activity 1 causes node 6 to be replaced by node 9 and node 7 to be replaced by node 8. The zero indicates no further modifications are caused by activity 1. Next, the realization of activity 8 causes node 3 to be replaced by node 4. Because of the method for storing these modifications, only an approximate upper limit can be given on the number of modifications permitted.

Error Messages

GERTS II has included tests for illogical conditions in many of the subprograms. Errors of this type should not occur, but if they do, a probable cause would be errors in the data input or incorrect dimensioning of arrays (or exceeding the dimension of an array). Figure 4 presents a list of error codes, the subroutine in which the error is located, and a possible cause for the error. Figure 5 is an illustration of an error exit obtained from GERTS II. In addition to the error code, the time that the error occurred is also printed. The filing array, NSET, that stores all activities and end of activity events is printed. The array QSET is a floating point array corresponding to NSET and stores either the event time or the probability associated with an activity. Next the array SUMA is printed which contains the statistical storage areas for statistics collected during the program. Time statistics and counter statistics for each node are stored consecutively. The column headings are the row number in the array SUMA, the sum of the times or counts in all simulation runs up to the error time, the sum of the squares of these variables up to the error time, the number of runs in which statistics were collected (this may not be the total number of runs since a node is not necessarily realized in every run), and the minimum and maximum values observed up to the current run. The last array printed is the array containing histogram information that has been collected up to the error time.

ERROR CODE	SUBROUTINE	CONDITION
22	GASP	The number of entries in a file is negative. The probable cause is incorrect dimensions.
23	SCHAT	An error has been identified in the pointing system which relates entries in a file. Probable cause, would be incorrect dimensions.
87	FILEM	All the storage space for activities and end of activity events has been allocated. The value in Field 8 of Data Card 1 should be increased. If it exceeds 1000, then the dimensions of NSET and QSET in the main program should be increased accordingly. The dimension of QSET should be set at the maximum number of activities and simultaneous end of activity events. The dimension of NSET should be 7 times this number.
88	FILEM	A negative value for a pointer has been detected. The possible error involves incorrect dimensions.
90	COLCT	Number of sink nodes, number of counters or the combination of the two is incorrect.
95	DATAN	A start node has been given a negative number, or the number of sinks and/or the number of counters is incorrectly specified.
97	RMOVE	A possible cause is incorrect dimensions.
98	SUMRY	Incorrect value for number of counts or number of sink nodes.

Figure 4. Error Codes Associated with GERT Simulation Program II

ERROR EXIT, TYPE 97 ERROR.

FILE STATUS AT TIME 9697.0669

NSET

1	0	0	0	0	0	142	9999
2	4	1	1	0	0	3	9999
3	27	1	1	0	1	7777	2
4	5	2	1	0	3	11	9999
5	6	2	1	0	4	7777	9999
6	7	3	1	0	3	7777	9999
7	8	2	1	0	4	7777	9999
8	9	3	1	0	3	7777	9999
9	10	2	1	0	4	7777	9999
10	11	3	1	0	0	7777	9999
11	12	5	4	0	0	7777	4
12	13	1	1	0	0	13	9999
13	14	1	1	0	0	7777	12
14	15	5	4	0	0	7777	9999
15	15	5	4	0	0	7777	9999
16	16	1	1	0	0	17	9999
17	17	1	1	0	0	7777	16
18	18	5	4	0	0	7777	9999
19	18	5	4	0	0	7777	9999
20	19	1	1	0	0	21	9999

QSET

1	0.000000E 00
2	0.600000E 00
3	0.100000E 01
4	0.100000E 01
5	0.100000E 01
6	0.100000E 01
7	0.100000E 01
8	0.100000E 01
9	0.100000E 01
10	0.100000E 01
11	0.100000E 01
12	0.700000E 00
13	0.100000E 01
14	0.100000E 01
15	0.100000E 01
16	0.700000E 00
17	0.100000E 01
18	0.100000E 01
19	0.100000E 01
20	0.700000E 00

ARRAY SUMA

1	0.0000	0.0000	1.0000	0.0000	0.0000
2	0.0000	0.0000	1.0000	0.0000	0.0000
3	0.0000	0.0000	1.0000	0.0000	0.0000
4	0.0000	0.0000	1.0000	0.0000	0.0000

ARRAY JCELS

1	1	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	

Figure 5. Example of a GERT Simulation Error Output

EXAMPLES

In this section examples of the use of the GERTS II are presented. The purpose of the examples is to illustrate how the concepts contained within GERT can be simulated. Details regarding the development of the network and the relation of the network to actual problems will not be included since the networks have been constructed to illustrate the use of the simulation program and not the problem solving procedure of GERT. For each example the following will be given:

(1) The GERT network diagram

(2) The data input to the simulation program

(3) The output from GERTS II, which includes the network description, activity parameters, network modifications, a description of the network characteristics in readable form and the final results from the simulation.

The examples will be labeled as projects 11 through 16 to agree with the numbers used in the simulations and to avoid confusion with the examples given in previous reports.

Project 11: Modifying A Network Based on Elapsed Time.

The network for project 11 is shown in Figure 6. Numbers shown in squares on the figure indicate activity numbers and are used to illustrate potential modifications to the network. The activity numbers are interpreted as described previously. Thus, when activity 1, a branch between nodes 2 and 5 is realized, the network is modified by having node 3 replaced by node 6. The dashed branch between node 3 and 6 labeled with a 1 in a square indicates the modification to occur when activity 1 is realized. In Figure 6, the branches from node 2 to 3, 3 to 4, and 6 to 4 are tagged with counters of type 1. That is, in the realization of the network, every time one of these three branches is realized, the count of type 1 is indexed by 1. The branch from node 6 to node 6 is tagged with a counter of type 2. Node 2 has a deterministic output so that both branches emanating from node 2 are taken. The branch from node 2 to node 5 is used to represent a clock. When the branch is realized, the clock time equals the elapsed time to traverse the branch from node 2 to node 5. When this occurs, the network is modified. If a

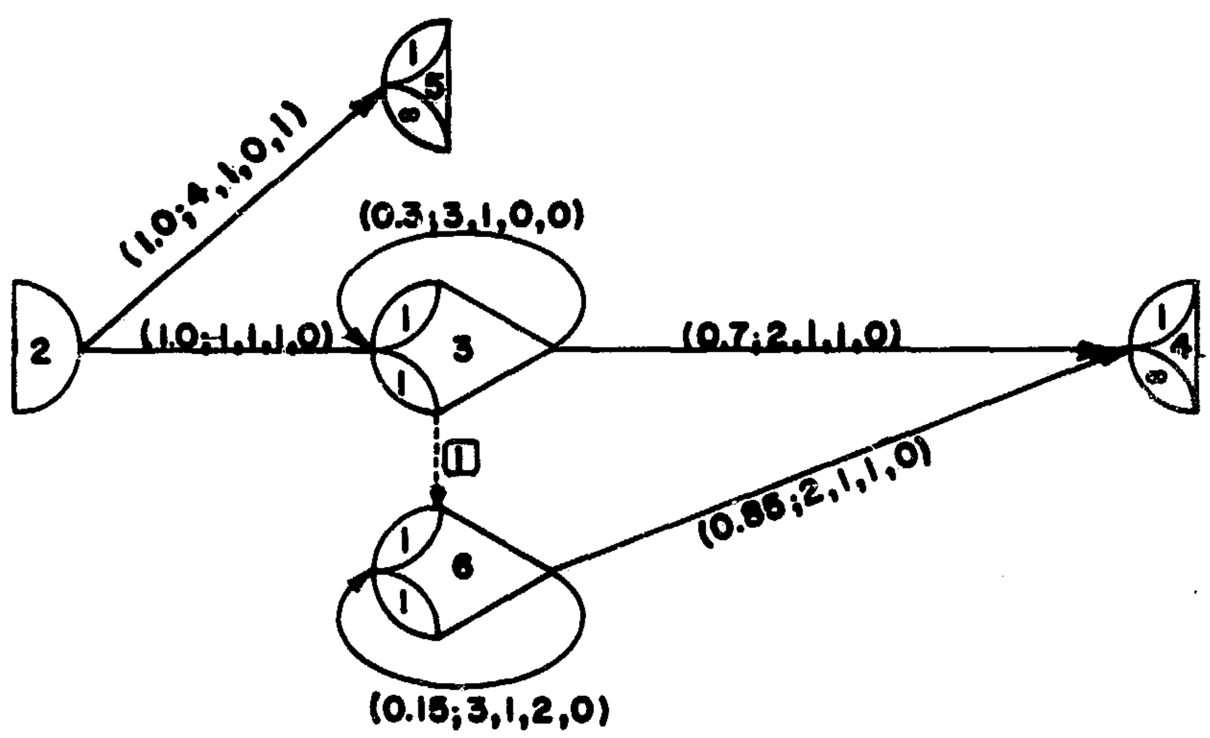


Figure 6. The GERT Network for Project 11

network was to be modified at many points in time, this would be accomplished by having a path of activities with each branch having a time parameter equal to the time between the desired network modification times. For project 11 all times will be considered as constants with the parameter number assigned to a branch equal to the constant time parameter.

The first output from GERTS II consists of the network description, activity parameters and network modifications. These are shown in Figure 7. In the network description section of Figure 7, the parameters associated with each branch are specified, and it is seen that the time required to go from node 2 to node 3 is one time unit (recall distribution type 1 indicates a constant time and we have set the parameter number equal to the constant value for this example). In this problem only the branch from node 2 to node 5 need by given an activity number since it involves a potential network modification. However, if for identification purposes it is desirable to give each branch of the network an activity number, it is permissible to do so. Only those activities specified in the network modification section will cause the network to be modified. Activity numbers which are not specified as causing a network modification do not affect the simulation process. The network modification section of Figure 7 gives the activity number whose realization causes the network to be changed and then the pairs of nodes involved in the network modification. Under each column headed by the word "Node" is the number of the node to be deleted from the network. Under the heading descriptor "File" is the node to be inserted into the network. (The word "File"

GERT SIMULATION PROJECT 11 BY PRITSKER
DATE 3/ 13/ 1969

NETWORK DESCRIPTION

START NODE	END NODE	PARAMETER NUMBER	DISTRIBUTION TYPE	COUNT TYPE	ACTIVITY NUMBER	PROBABILITY
2	3	1	1	1	0	1.0000
2	5	4	1	0	1	1.0000
3	4	2	1	1	0	0.7000
3	3	3	1	0	0	0.3000
6	4	2	1	1	0	0.8500
6	6	3	1	2	0	0.1500

ACTIVITY PARAMETERS

PARAMETER NUMBER	PARAMETERS 1	2	3	4
1	1.0000	0.0000	0.0000	0.0000
2	2.0000	0.0000	0.0000	0.0000
3	3.0000	0.0000	0.0000	0.0000
4	4.0000	0.0000	0.0000	0.0000

NETWORK MODIFICATIONS

ACTIVITY	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE
1	3	6																		

Figure 7. GERT Simulation Output Showing Network Description, Activity Parameters, and Network Modifications

Card data		Identification
PRITSKER 11 03 13 1969 0400 4 30 1267	1	DA11 10
6 1 1 1 1 2 1	2	DA11 20
2 4	3	DA11 30
0 1 1 1 1	4	DA11 40
1 4 1 1 4	5	DA11 50
3.0	6	DA11 60
3.0	7	DA11 70
1.0	8	DA11 80
2.0		DA11 90
3.0		DA11 100
4.0		DA11 110
1.0 2 3 1 1 1	9	DA11 120
1.0 2 5 4 1 0 1		DA11 130
0.7 3 4 2 1 1		DA11 140
.30 3 3 3 1 0		DA11 150
0.85 6 4 2 1 1		DA11 160
0.15 6 6 3 1 2		DA11 170
0		DA11 180
1 3 6	10	DA11 190
0		DA11 200

Figure 8. Listing of Data Input for Project 11

evolved from the method employed in the simulation program for incorporating network modifications). Each node-file pair represents a network modification caused by the activity number specified in the row. An activity can effect up to ten network modifications.

The probability of taking a particular branch in a simulation of the network is shown in the last column under "Network Description." In the data input the probabilities are inserted as the first field for each activity as shown in Figure 8 which is the listing of the input cards for project 11.

The second output from the simulation program is shown in Figure 9. In this figure, a description is presented in a readable form of the significant characteristics of the network and the nodes of the network. Two values are shown for the initial random number since it can be either a fixed or floating point value depending on the computer employed. The initial random number, ISEED, is recorded as a fixed point variable that was equal to 1267. Since the floating point initial random number was not used, a zero is printed.

The final results for 400 simulations are presented in Figure 10. The statistics associated with the time a node is realized and the number of counts of each type that occurred before the node was realized are presented. For project 11, it is seen that node 4 is realized in all 400 simulations and that the average time of realization is 3.7725 time units with a minimum of 3 and a maximum of 12 time units. As expected, in every simulation two counters of type 1 were realized prior to the realization of node 4. This can be seen from the output since the mean,

HIGHEST NODE NUMBER IS 6
NUMBER OF SOURCE NODES IS 1
NUMBER OF SINK NODES IS 1
NUMBER OF NODES TO REALIZE THE NETWORK IS 1
STATISTICS COLLECTED ON 1 NODES
NUMBER OF PARAMETER SETS IS 4
INITIAL RANDOM NUMBER IS 1267 0.0000

SOURCE NODE NUMBERS
2

SINK NODE NUMBERS
4

STATISTICS COLLECTED ALSO ON NODES

NODE	NUMBER RELEASES	NODE TYPE
2	0	1
3	1	4
4	1	1
5	1	1
6	1	4

Figure 9. Description of the Significant Characteristics of the Network for Project 11

GERT SIMULATION PROJECT 11 BY PRITSKER
DATE 3/ 13/ 1969

FINAL RESULTS FOR 400 SIMULATIONS

NODE	PROB./COUNT		MEAN	STD.DEV.	MIN.	MAX.
4	1.0000		3.7725	1.5055	3.0000	12.0000
4		1	2.0000	0.0000	2.0000	2.0000
4		2	0.0275	0.1784	0.0000	2.0000

HISTOGRAMS

NODE	LOWER LIMIT	CELL WIDTH	FREQUENCIES										
4	3.00	3.00	0	308	82	9	1	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	

Figure 10. Final Results for Project 11 from GERT Simulation Program

min and max are two and the standard deviation is zero. Counter type 2 identifies the number of times the self loop around node 6 was realized. From the output it is seen that on the average the self loop around node 6 was taken only 2.75% of the time and at most twice in any one simulation run.

Since the network for project 11 is small, it is possible to reformulate the network in a form which contains only EXCLUSIVE-OR nodes or their equivalent. The reformulated network is shown in Figure 11.

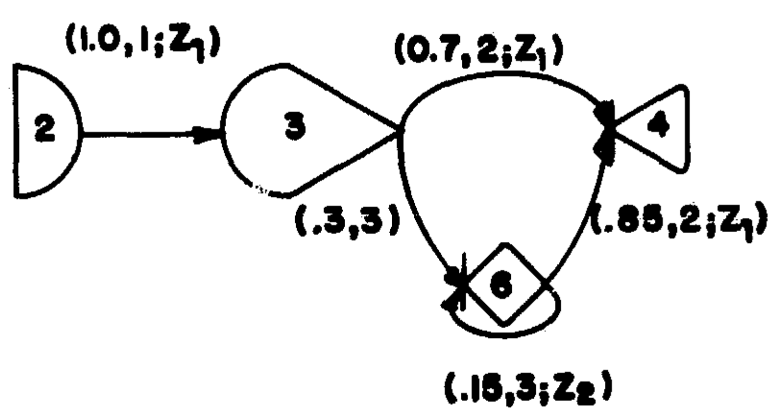


Figure 11. Network of Project 11 in terms of EXCLUSIVE-OR nodes

For this network the w-generating function [13] for the one branch equivalent network is

$$w(s,z_1,z_2) = .7e^{3s}z_1^2 + \frac{.255e^{6s}z_1^2}{(1-.15e^{3s}z_2)}$$

$$= z_1^2 \, [.7e^{3s} + .255e^{6s} \sum_{j=o}^{\infty} (.15e^{3s}z_2)^j]$$

From the above, it is found that the expected time to realize node 4 is 4.0588 and the expected number of times the self loop around node 6 is taken is 0.053. Comparision between the theoretical results and the simulation results verify that GERTS II is performing adequately. The difference between the theoretical value and simulation value for the average number of counts of type 2 is due to the low probability associated with obtaining a higher number of traversals of the self loop. For example, the probability of traversing the self loop of node 6 twice in a given realization of the network is 0.0057 and for three traversals is 0.00086. In 400 simulations of the network we would expect only two runs with traversals of the self loop. The histogram data shown in Figure 10 indicates that node 4 was realized in 12 time units one time. This indicates that the self loop was taken twice in one run. The histogram also indicates that no higher order traversals of the self loop were obtained.

Project 12: Illustration of New Node Types

Figure 12 is a small network to illustrate node type 8. Node 3 of Figure 12 is a type 8 node which means that as soon as one of the activities incident to the node is realized all other activities which are in process and end at node 3 are halted (cancelled). In simulation parlance halting an activity involves removing the scheduled end of activity event from the calendar of events to occur. Node 3 has an even node number (type 8) since its output is probabilistic. If its output side was deterministic it would be a node type 7. Since node 3 can be realized more than once it cannot be a type 2 node. It cannot be a type 4 node because when the node is realized other end of activity events for activities incident on the node should not cause the node to be realized again. Thus, the need for node type 8.

The listing of the input data for project 12 is shown in Figure 13. The network description as obtained from GERTS II simulation program is given in Figure 14. Each branch from node 2 to node 3 is taken with probability one (node 2 is type 3 which indicates that node 2 may be realized many times and its output side is deterministic). The time associated with each branch from node 2 to 3 is exponentially distributed with mean times of 6, 8 and 10 time units. The time to the first realization of any of these activities in a particular simulation run will also be exponentially distributed since the minimum of exponentially distributed random variables is also exponentially distributed. For project 12 the theoretical mean time until the first realization of node 3 is 2.55 time units. To indicate the sequence of events that occur during the simulation of the network of project 12, the trace feature of GERTS II was used. (Data card 11 is not shown in Figure 13 as the trace was run on IBM 1800 on an earlier run.) Figure 15 illustrates the results of the trace of events that occurred during simulation runs number 5-10. In Figure 15,

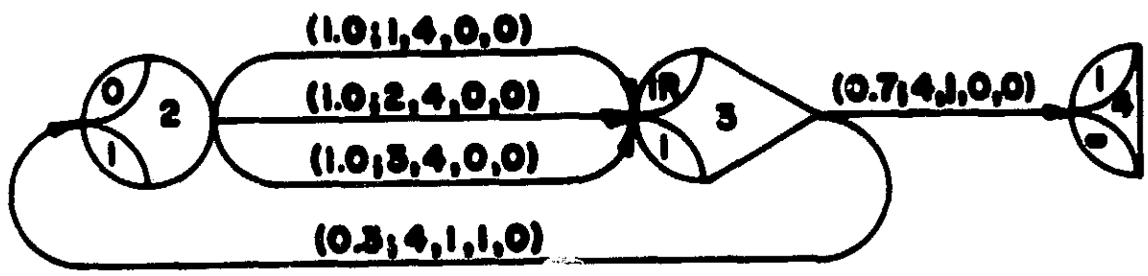


Figure 12. The GERT Network for Project 12

PRITSKER 12 04 28 1969 400 4 20 1267	1	DA12 10
4 1 1 1 2 1 0	2	DA12 20
2 4 3	3	DA12 30
1 1 1	4	DA12 40
3 8 1	5	DA12 50
4. 0 2.0	6	DA 12 60
2.0 1.0	7	DA12 70
6.0 0.0 100.0 1.0	8	DA12 80
8.0 0.0 100.0 1.0		DA12 90
10.0 0.0 100.0 1.0		DA12 100
1.0		DA12 110
1.0 2 3 1 4	9	DA12 120
1.0 2 3 2 4		DA12 130
1.0 2 3 3 4		DA12 140
0.7 3 4 4 1		DA12 150
0.30 3 2 4 1 1		DA12 160
0		DA12 170

Figure 13. Listing of Data Input for Project 12

GERT SIMULATION PROJECT 12 BY PRITSKER
DATE 4/ 28/ 1969

NETWORK DESCRIPTION

START NODE	END NODE	PARAMETER NUMBER	DISTRIBUTION TYPE	COUNT TYPE	ACTIVITY NUMBER	PROBABILITY
2	3	1	4	0	0	1.0000
2	3	2	4	0	0	1.0000
2	3	3	4	0	0	1.0000
3	4	4	1	0	0	0.7000
3	2	4	1	1	0	0.3000

ACTIVITY PARAMETERS

PARAMETER NUMBER	PARAMETERS 1	2	3	4
1	6.0000	0.0000	100.0000	1.0000
2	8.0000	0.0000	100.0000	1.0000
3	10.0000	0.0000	100.0000	1.0000
4	1.0000	0.0000	0.0000	0.0000

HIGHEST NODE NUMBER IS 4
NUMBER OF SOURCE NODES IS 1
NUMBER OF SINK NODES IS 1
NUMBER OF NODES TO REALIZE THE NETWORK IS 1
STATISTICS COLLECTED ON 2 NODES
NUMBER OF PARAMETER SETS IS 4
INITIAL RANDOM NUMBER IS 1267 0.0000

SOURCE NODE NUMBERS
2

SINK NODE NUMBERS
4

STATISTICS COLLECTED ALSO ON NODES
3

NODE	NUMBER RELEASES	NODE TYPE
2	1	3
3	1	8
4	1	1

Figure 14. Echo Check of Characteristics of GERT Network for Project 12 Obtained from GERT Simulation Program

the attributes associated with the activity are the parameter number, distribution type, count type and activity number. In this example no activity number is associated with a branch so that the identification of each activity that occurred during tracing must be determined by the attributes of the activity. The results obtained from GERTS II are presented in Figure 16. Statistics were collected for two nodes in this simulation. The output will always present the statistics on the sink nodes first. For project 12 the sink node was node 4 which was realized in all simulations and had an average time of realization of 4.8964. In the simulation, a counter was used to count the number of times the feedback branch from node 3 to node 2 was realized. From the output it is seen that the average number of times the branch from node 3 to node 2 was taken is 0.4075. Note that this is not the probability of taking the branch from node 3 to node 2 but is the average number of times in a simulation run that this branch is taken. For project 12 it is seen that in one simulation of the network the branch from node 3 to node 2 was taken seven times. As discussed previously, a count is only obtained if the activity tagged is realized prior to the realization of the node for which statistics are being collected. Thus, in the statistics for node 3 all statistics associated with the counter are zero, since the branch from node 3 to node 2 cannot be realized prior to the first realization of node 3. The average time to reach node 3 is seen to be 2.5800 time units and the standard deviation to be 2.5049 time units. These values and the histogram for node 3 verify that the simulation is reproducing the

AT TIME	0.37	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	0 WAS REALIZED
AT TIME	1.37	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	0 WAS REALIZED
AT TIME	0.40	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	0 WAS REALIZED
AT TIME	1.40	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	0 WAS REALIZED
AT TIME	3.17	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	0 WAS REALIZED
AT TIME	4.17	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	0 WAS REALIZED
AT TIME	11.97	ACTIVITY ON NODE	3	WITH ATTRIBUTES	3	4	0	0 WAS REALIZED
AT TIME	12.97	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	0 WAS REALIZED
AT TIME	0.86	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	0 WAS REALIZED
AT TIME	1.86	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	0 WAS REALIZED
AT TIME	4.93	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	0 WAS REALIZED
AT TIME	5.93	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	0 WAS REALIZED
AT TIME	2.88	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	0 WAS REALIZED
AT TIME	3.88	ACTIVITY ON NODE	2	WITH ATTRIBUTES	4	1	1	0 WAS REALIZED
AT TIME	4.02	ACTIVITY ON NODE	3	WITH ATTRIBUTES	3	4	0	0 WAS REALIZED
AT TIME	5.02	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	0 WAS REALIZED

Figure 15. Trace of the Realization of Activities for Project 12

GERT SIMULATION PROJECT 12 BY PRITSKER
DATE 4/ 28/ 1969

FINAL RESULTS FOR 400 SIMULATIONS

NODE	PROB./COUNT		MEAN	STD.DEV.	MIN.	MAX.
4	1.0000		4.8964	3.7915	1.0007	25.9128
4		1	0.4075	0.7925	0.0000	7.0000
3	1.0000		2.5800	2.5049	0.0007	16.2502
3		1	0.0000	0.0000	0.0000	0.0000

HISTOGRAMS

NODE	LOWER LIMIT	CELL WIDTH	FREQUENCIES									
4	4.00	2.00	204	77	56	24	17	8	7	2	3	0	1
			1	0	0	0	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	
3	2.00	1.00	210	61	44	21	19	20	10	4	5	3	0
			1	0	1	0	1	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	

Figure 16. Final Results for Simulation of Project 12

exponential distribution which as was discussed earlier is known to be the distribution of the time until the first realization of node 3.

Project 12 was modified to project 121 by making node 3 a type 6 node* and setting the number of releases to three, i.e., an AND node. Figure 17 illustrates the network. The only changes required in the input data are in data card types 4 and 5 where the number of releases and node type for node 3 are changed to 3 and 6 respectively. The output from GERTS II as run on the IBM 1800 is shown in Figures 18, 19 and 20. Project 12 was further modified to create project 122 by making the number of releases for node 3 equal to 2. Since three branches are incident to node 3 it will be made a type 8 node so that after two of the branches are realized, the third activity will be cancelled. In Figure 21, a trace of project 122 is shown and illustrates the operation of a node type 8 which in this case involves realizing two out of three branches. For this project, each activity is given an activity number to simplify identifying activity realizations in the trace. Note no network modifications are included in the project, therefore the activity numbers are used only for identification purposes.

Project 13: Simulation of a PERT Network

Project 13 is identical to Example 3 of the first version of the GERT Simulation Program.[7] It is included to allow comparisons between

*Actually node 3 could be considered as a type 8 node as long as the number of releases is set to 3. The use of the type 6 node saves computer execution time since there is no need to search for scheduled end-of-activity events incident on node 3 when it is realized.

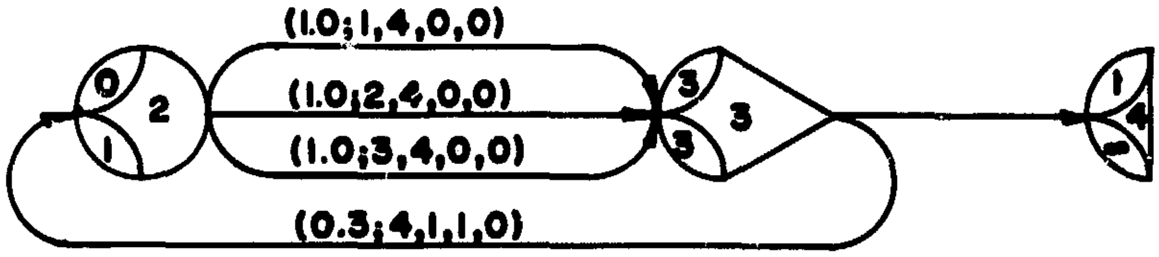


Figure 17. The GERT Network for Project 121

GERT SIMULATION PROJECT 121 BY PRITSKER
DATE 4/ 28/ 1969

NETWORK DESCRIPTION

START NODE	END NODE	PARAMETER NUMBER	DISTRIBUTION TYPE	COUNT TYPE	ACTIVITY NUMBER	PROBABILITY
2	3	1	4	0	0	1.0000
2	3	2	4	0	0	1.0000
2	3	3	4	0	0	1.0000
3	4	4	1	0	0	0.7000
3	2	4	1	1	0	0.3000

ACTIVITY PARAMETERS

PARAMETER NUMBER	PARAMETERS 1	2	3	4
1	6.0000	0.0000	100.0000	1.0000
2	8.0000	0.0000	100.0000	1.0000
3	10.0000	0.0000	100.0000	1.0000
4	1.0000	0.0000	0.0000	0.0000

Figure 18. Network Description and Activity Parameters for Project 121

```
HIGHEST NODE NUMBER IS      4
NUMBER OF SOURCE  NODES IS     1
NUMBER OF SINK NODES IS     1
NUMBER OF NODES TO REALIZE THE NETWORK IS     1
STATISTICS COLLECTED ON     2 NODES
NUMBER OF PARAMETER SETS IS     4
INITIAL RANDOM NUMBER IS 1267     0.0000

SOURCE NODE NUMBERS
      2

SINK NODE NUMBERS
      4

STATISTICS COLLECTED ALSO ON NODES
      3
```

NODE	NUMBER RELEASES	NODE TYPE
2	1	3
3	3	6
4	1	1

Figure 19. Significant Network Characteristics for Project 121

GERT SIMULATION PROJECT 121 BY PRITSKER
DATE 4/ 28/ 1969

FINAL RESULTS FOR 400 SIMULATIONS

NODE	PROB./COUNT		MEAN	STD.DEV.	MIN.	MAX.
4	1.0000		22.2289	15.9343	2.8852	104.6472
4		1	0.4375	0.7632	0.0000	4.0000
3	1.0000		14.4575	9.0487	0.5447	53.4688
3		1	0.0000	0.0000	0.0000	0.0000

HISTOGRAMS

NODE	LOWER LIMIT	CELL WIDTH	FREQUENCIES										
4	4.	2.0	9	20	25	34	34	32	22	25	20	14	20
			23	9	15	18	5	9	14	8	5	5	4
			3	3	2	2	2	5	1	2	1	9	
3	2.	1.0	4	10	11	14	19	25	22	21	27	18	25
			20	13	15	18	17	15	8	7	6	5	11
			11	8	5	3	7	5	4	4	3	19	

Figure 20. Final Results of Simulation of Project 121

AT TIME	0.79	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	0.99	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	1	WAS REALIZED
AT TIME	1.99	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	4	WAS REALIZED
AT TIME	0.04	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	2.04	ACTIVITY ON NODE	3	WITH ATTRIBUTES	3	4	0	3	WAS REALIZED
AT TIME	3.04	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	4	WAS REALIZED
AT TIME	4.48	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	5.61	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	1	WAS REALIZED
AT TIME	6.61	ACTIVITY ON NODE	2	WITH ATTRIBUTES	4	1	1	5	WAS REALIZED
AT TIME	10.35	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	13.67	ACTIVITY ON NODE	3	WITH ATTRIBUTES	3	4	0	3	WAS REALIZED
AT TIME	14.67	ACTIVITY ON NODE	2	WITH ATTRIBUTES	4	1	1	5	WAS REALIZED
AT TIME	17.62	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	19.75	ACTIVITY ON NODE	3	WITH ATTRIBUTES	3	4	0	3	WAS REALIZED
AT TIME	20.75	ACTIVITY ON NODE	2	WITH ATTRIBUTES	4	1	1	5	WAS REALIZED
AT TIME	21.00	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	1	WAS REALIZED
AT TIME	24.93	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	25.93	ACTIVITY ON NODE	2	WITH ATTRIBUTES	4	1	1	5	WAS REALIZED
AT TIME	28.32	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	1	WAS REALIZED
AT TIME	28.40	ACTIVITY ON NODE	3	WITH ATTRIBUTES	3	4	0	3	WAS REALIZED
AT TIME	29.40	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	4	WAS REALIZED
AT TIME	0.64	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	1	WAS REALIZED
AT TIME	5.49	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	6.49	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	4	WAS REALIZED
AT TIME	1.73	ACTIVITY ON NODE	3	WITH ATTRIBUTES	1	4	0	1	WAS REALIZED
AT TIME	11.58	ACTIVITY ON NODE	3	WITH ATTRIBUTES	2	4	0	2	WAS REALIZED
AT TIME	12.58	ACTIVITY ON NODE	4	WITH ATTRIBUTES	4	1	0	4	WAS REALIZED

Figure 21. Trace of Realization of Activities for Project 122

the first version of the GERT Simulation Program and GERTS II. The PERT network to be simulated is shown in Figure 22. The data input is shown

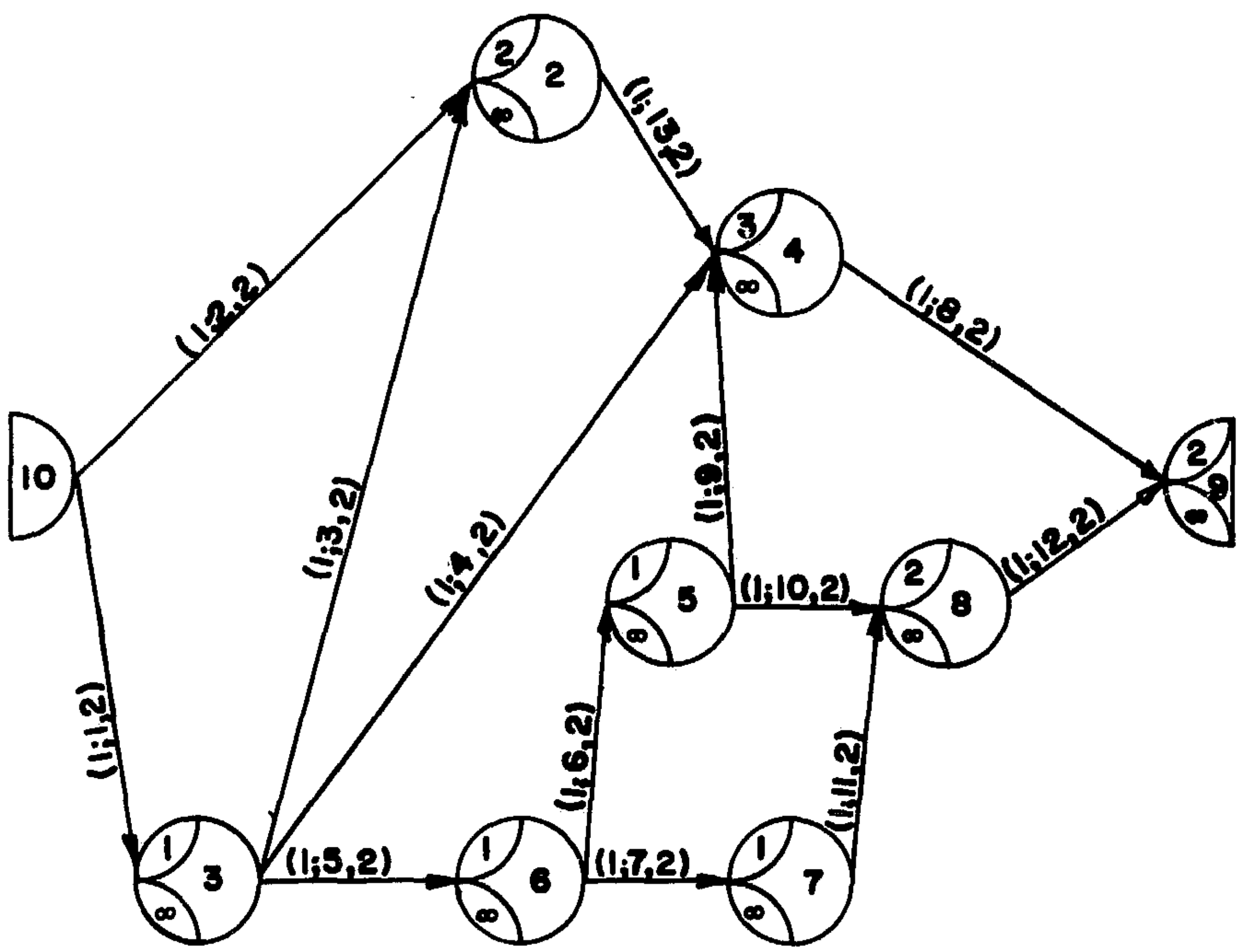


Figure 22. The PERT Network for Project 13

in Figure 23 and consists of 37 data cards as compared to 55 required before. The network description, activity parameters, and node characteristics are shown in Figures 24 and 25. From Figure 24 it is seen that all activity times have distribution type 2 (a normal distribution). No count or network modifications are included in this project in order that a direct comparison could be made with the previous simulation

PRITSKER	13	04	22	1969	0400	13	30	1567		1	DA13 10
10	1	1	1	5						2	DA13 20
10	9	5	6	7	8					3	DA13 30
2	1	3	1	1	1	2	2000			4	DA13 40
1	1	1	1	1	1	1	1	1	1	5	DA13 50
32.0	12.0	5.0	16.0	18.0						6	DA13 60
1.0	1.0	1.0	1.0	1.0						7	DA13 70
13.0	0.0	100.	3.00							8	DA13 80
5.5	0.0	100.	1.18								DA13 90
7.0	0.0	100.	1.00								DA13 100
16.5	0.0	100.	2.50								DA13 110
5.2	0.0	100.	0.84								DA13 120
6.0	0.0	100.	1.00								DA13 130
10.3	0.0	100.	1.67								DA13 140
3.2	0.0	100.	0.50								DA13 150
20.0	0.0	100.	3.32								DA13 160
4.0	0.0	100.	0.71								DA13 170
3.2	0.0	100.	0.84								DA13 180
16.5	0.0	100.	1.18								DA13 190
14.7	0.0	100.	1.34								DA13 200
1.0	2	4	13	2						9	DA13 210
1.0	5	8	10	2							DA13 220
1.0	3	2	3	2							DA13 230
1.0	3	4	4	2							DA13 240
1.0	3	6	5	2							DA13 250
1.0	4	9	8	2							DA13 260
1.0	5	4	9	2							DA13 270
1.0	6	5	6	2							DA13 280
1.0	6	7	7	2							DA13 290
1.0	7	8	11	2							DA13 300
1.0	8	9	12	2							DA13 310
1.0	10	3	2	2							DA13 320
1.0	10	2	1	2							DA13 330
	0										DA13 340

Figure 23. Listing of Data Input for Project 13

GERT SIMULATION PROJECT 13 BY PRITSKER
DATE 4/ 22/ 1969

NETWORK DESCRIPTION

START NODE	END NODE	PARAMETER NUMBER	DISTRIBUTION TYPE	COUNT TYPE	ACTIVITY NUMBER	PROBABILITY
2	4	13	2	0	0	1.0000
3	2	3	2	0	0	1.0000
3	4	4	2	0	0	1.0000
3	6	5	2	0	0	1.0000
4	9	8	2	0	0	1.0000
5	8	10	2	0	0	1.0000
5	4	9	2	0	0	1.0000
6	5	6	2	0	0	1.0000
6	7	7	2	0	0	1.0000
7	8	11	2	0	0	1.0000
8	9	12	2	0	0	1.0000
10	3	2	2	0	0	1.0000
10	2	1	2	0	0	1.0000

ACTIVITY PARAMETERS

PARAMETER NUMBER	PARAMETERS 1	2	3	4
1	13.0000	0.0000	100.0000	3.0000
2	5.5000	0.0000	100.0000	1.1800
3	7.0000	0.0000	100.0000	1.0000
4	16.5000	0.0000	100.0000	2.5000
5	5.2000	0.0000	100.0000	0.8400
6	6.0000	0.0000	100.0000	1.0000
7	10.3000	0.0000	100.0000	1.6700
8	3.2000	0.0000	100.0000	0.5000
9	20.0000	0.0000	100.0000	3.3200
10	4.0000	0.0000	100.0000	0.7100
11	3.2000	0.0000	100.0000	0.8400
12	16.5000	0.0000	100.0000	1.1800
13	14.7000	0.0000	100.0000	1.3400

Figure 24. Network Description and Activity Parameters for Project 13

```
HIGHEST NODE NUMBER IS    10
NUMBER OF SOURCE NODES IS     1
NUMBER OF SINK NODES IS     1
NUMBER OF NODES TO REALIZE THE NETWORK IS      1
STATISTICS COLLECTED ON     5 NODES
NUMBER OF PARAMETER SETS IS   13
INITIAL RANDOM NUMBER IS 1567     0.0000

SOURCE NODE NUMBERS
     10

SINK NODE NUMBERS
      9

STATISTICS COLLECTED ALSO ON NODES
      5    6    7    8
```

NODE	NUMBER RELEASES	NODE TYPE
2	2	1
3	1	1
4	3	1
5	1	1
6	1	1
7	1	1
8	2	1
9	2	1
10	0	1

Figure 25. Significant Network Characteristics for Project 13

program. The final statistics for Project 13 are given in Figure 26 for 400 simulations of the network.

Project 14. Specifying an Activity's Parameters Based on the Realization of the First of Two Activities

As an illustration of the use of the network modification features of GERTS II, Project 14 has been developed. The network for Project 14

GERT SIMULATION PROJECT 13 BY PRITSKER
DATE 4/ 22/ 1969

FINAL RESULTS FOR 400 SIMULATIONS

NODE	PROB./COUNT	MEAN	STD.DEV.	MIN.	MAX.
9	1.0000	41.7559	2.6706	33.8642	49.5449
5	1.0000	16.6011	1.7254	10.9789	20.7607
6	1.0000	10.6242	1.4076	5.8548	14.7819
7	1.0000	20.6619	2.2235	14.4070	27.2393
8	1.0000	23.9223	2.3380	17.7705	31.3513

HISTOGRAMS

NODE	LOWER LIMIT	CELL WIDTH	FREQUENCIES										
9	32.00	1.00	0	0	1	0	5	5	23	28	47	50	53
			59	47	34	22	16	5	4	1	0	0	0
			0	0	0	0	0	0	0	0	0	0	
5	12.00	1.00	2	2	17	55	72	94	72	49	30	7	0
			0	0	0	0	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	
6	5.00	1.00	0	1	1	16	31	83	104	96	53	14	1
			0	0	0	0	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	
7	16.00	1.00	7	13	26	38	62	83	64	47	35	15	6
			3	1	0	0	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	
8	18.00	1.00	2	4	12	23	40	64	65	68	41	38	29
			8	4	1	1	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	

Figure 26. Final Results of Simulation for Project 13

is shown in Figure 27. Nodes 5 and 8 represent the same milestone. It is desired that the activity between 5 and 7 (or equivalently 8 and 7) be dependent on which activity causes node 5 to be realized, i.e., branch 2-5 or 3-5. The method for accomplishing this by network modifications is described below.

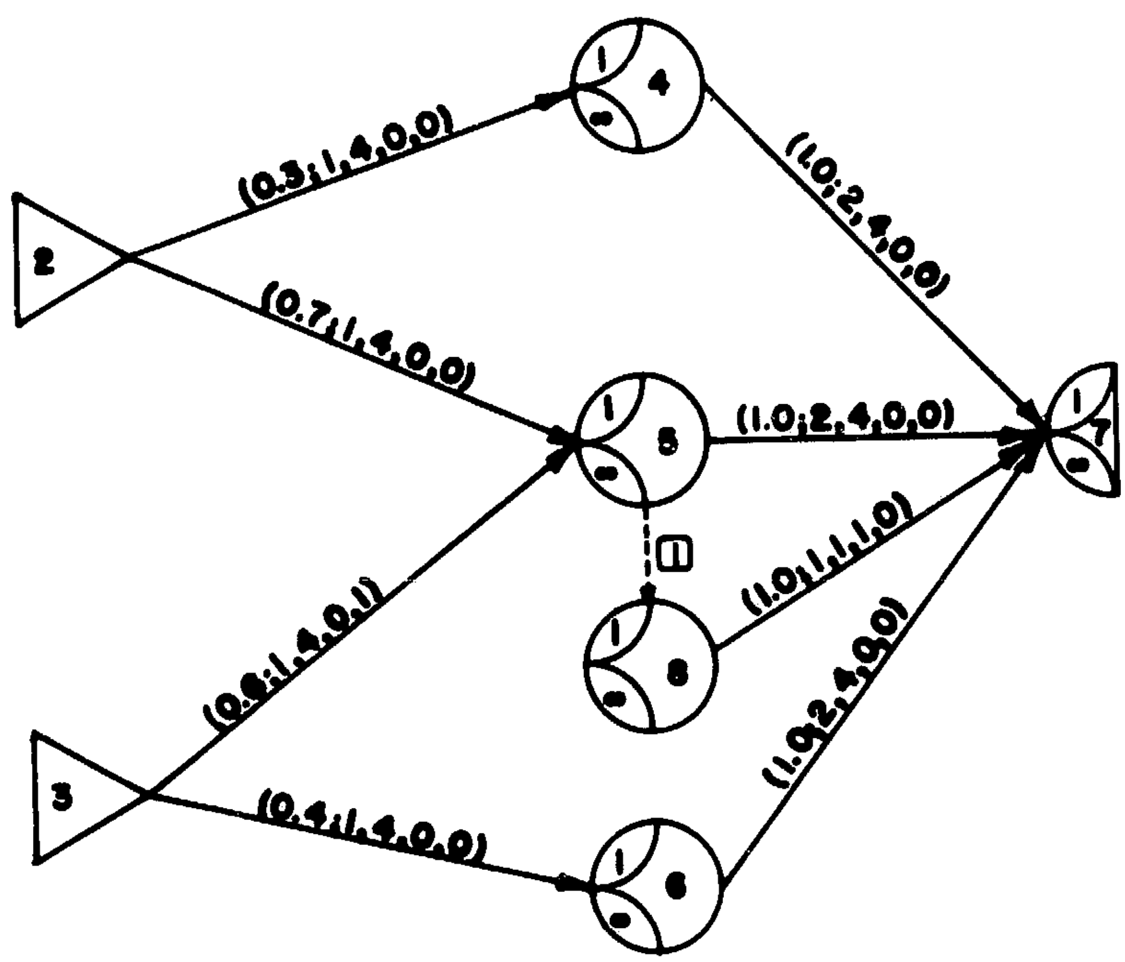


Figure 27. GERT Network for Project 14

There are two source nodes associated with the network: nodes 2 and 3. From node 2, either the activity to node 4, or the activity to node 5 is taken. Therefore, node 2 is a probabilistic node. Similarly, from node 3 an activity leads to node 5 or node 6 and node 3 is also a probabilistic node. The activity from node 3 to node 5 is defined as activity 1. If activity 1 is completed before the activity going from node 2 to node 5, node 5 is replaced node 8 in the network. The activity from node 8 to node 7 ha_ associated with it the characteristics of going from node 5 to node 7 if activity 1 is realized before the activity from node 2 to node 5. The activity from node 5 to node 7 has associated with it the characteristics of the activity when the activity from node 2 to node 5 is realized before activity 1. Both nodes 5 and 8 are type 1 nodes, i.e., they can only be realized once and their output type is deterministic (since there is only one activity emanating from the node, the simulation is faster if the node is defined as deterministic). Branches from nodes 4 and 6 are also incident to node 7 so that the completion of the network is determined by the realization of the first branch leading into node 7. The input data for project 14 is shown in Figure 28, and the description obtained from the GERT simulation program of the network is shown in Figures 29 and 30. The times associated with the activities of the network are Erlang distributed except for the activity from node 8 to node 7. The time for this activity is a constant ten time units since it has been assumed that if activity 1 causes node 5 to be realized, the follow-on activity can be done in a given time. If an activity with distribution type 2 also uses parameter set 1 then the time to perform the activity is characterized by the exponential distribution (the exponential is a special case of the Erlang distribution with parameter 4 equal to 1). For the network presented in Figure 27,

PRITSKER	14	4	25	1969	400	2	30	0267	1	DA14 10
8	2	1	1	2	1	1			2	DA14 20
2	3	7	5						3	DA14 30
0	0	1	1	1	1	1			4	DA14 40
2	2	1	1	1	0	1			5	DA14 50
2.0	2.0								6	DA14 60
1.0	1.0								7	DA14 70
10.0	0.0	100.0	1.0						8	DA14 80
5.0	0.0	100.0	2.0							DA14 90
0.7	2	5	1	4					9	DA14 100
0.30	2	4	1	4						DA14 110
0.6	3	5	1	4	0	1				DA14 120
0.40	3	6	1	4						DA14 130
1.0	4	7	2	4						DA14 140
1.0	5	7	2	4						DA14 150
1.0	6	7	2	4						DA14 160
1.0	8	[illegible]	1	1	1					DA14 170
	0									DA14 180
1	5	8	0						10	DA14 190
0										DA14 200

Figure 28. Listing of Input Data for Project 14

GERT SIMULATION PROJECT 14 BY PRITSKER
DATE 4/ 25/ 1969

NETWORK DESCRIPTION

START NODE	END NODE	PARAMETER NUMBER	DISTRIBUTION TYPE	COUNT TYPE	ACTIVITY NUMBER	PROBABILITY
2	5	1	4	0	0	0.7000
2	4	1	4	0	0	0.3000
3	5	1	4	0	1	0.6000
3	6	1	4	0	0	0.4000
4	7	2	4	0	0	1.0000
5	7	2	4	0	0	1.0000
6	7	2	4	0	0	1.0000
8	7	1	1	1	0	1.0000

ACTIVITY PARAMETERS

PARAMETER NUMBER	PARAMETERS 1	2	3	4
1	10.0000	0.0000	100.0000	1.0000
2	5.0000	0.0000	100.0000	2.0000

NETWORK MODIFICATIONS

ACTIVITY	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE
1	5	8																		

Figure 29. Network Description, Activity Parameters, and Network Modifications for Project 14

HIGHEST NODE NUMBER IS 8
NUMBER OF SOURCE NODES IS 2
NUMBER OF SINK NODES IS 1
NUMBER OF NODES TO REALIZE THE NETWORK IS 1
STATISTICS COLLECTED ON 2 NODES
NUMBER OF PARAMETER SETS IS 2
INITIAL RANDOM NUMBER IS 267 0.0000

SOURCE NODE NUMBERS
2 3

SINK NODE NUMBERS
7

STATISTICS COLLECTED ALSO ON NODES
5

NODE	NUMBER RELEASES	NODE TYPE
2	0	2
3	0	2
4	1	1
5	1	1
6	1	1
7	1	0
8	1	1

Figure 30. Significant Network Characteristics for Project 14

the possible values for the time to realize node 7, T_7 are;

Possible Expressions for T_7	Probability that T_7 Takes on This Expression
$\min [T_{24}+T_{47}; T_{36}+T_{67}]$	0.12
$\min [T_{24}+T_{47}; T_{35}+T_{87}]$	0.18
$\min [T_{25}+T_{57}; T_{36}+T_{67}]$	0.28
$T'_{25}+T_{57}$	0.21
$T'_{35}+T_{87}$	0.21

where

$$T'_{25} = T_{25} | T_{25} < T_{35}$$

and

$$T'_{35} = T_{35} | T_{25} \geq T_{35}.$$

We have

$$P[T'_{25} \leq t] = 1 - e^{-t/5},$$

$$P[T'_{35} \leq t] = 1 - e^{-t/5},$$

and

$$P[T_{25} < T_{35}] = \frac{1}{2}.$$

Even for this small problem, it is not easy to calculate the probability that the activity from node 8 to node 7 is the activity that causes node 7 to be realized. To obtain this information from the simulation program, a counter is tagged to the activity from node 8 to node 7. The the average number of times that this tagged activity is realized prior to the realization of node 7 is an estimate of the desired probability.

The final results from GERTS II for Project 14 are presented in Figure 31. The average time to reach node 7 is 14.0840 time units with a standard deviation of 6.8476 time units. The minimum and maximum values are 0.8367 and 46.8011. The large variation in the time to realize node 7 is due to the basic uncertainty in the times to perform each activity, i.e., the activity time is a random variable. In addition, uncertainty exists with regard to which path of the network will be involved in causing node 7 to be realized. Knowledge of this wide variability in the time to realize a project is important. The network modification employed in this project can also be thought of as the inclusion of a decision element which specifies the choice to be made based on the realization of a specific event.

From Figure 31 it can be seen that in 29.25% of the simulations (mean number of times counter I was realized), the activity from node 8 to node 7 caused node 7 to be realized. In 79.25% of the realizations of the network, node 5 was realized. There are two causes for not realizing node 5: (1) both activities 2-4 and 3-6 were selected; and (2) node 7 can be realized prior to the time that node 5 is realized. The latter case obtains, for example, when the sum of the times to go from node 3 to node 6 to node 7 is less than the time to go from node 2 to node 5. When node 5 is realized, the average completion time was 5.7346 time units. The time to realize node 5 is distributed according to the hyperexponential distribution since if the activity from node 2 to node 5 is taken and activity from node 3 to node 5 is not taken, we have an exponential distribution; if the activity from node 3 to node 5 is taken and the activity

GERT SIMULATION PROJECT 14 BY PRITSKER
DATE 4/ 25/ 1969

FINAL RESULTS FOR 400 SIMULATIONS

NODE	PROB./COUNT		MEAN	STD.DEV.	MIN.	MAX.
7	1.0000		14.0840	6.8476	0.8367	46.8011
7		1	0.2925	0.4555	0.0000	1.0000
5	0.7925		5.7346	5.1392	0.0018	36.1078
5		1	0.0000	0.0000	0.0000	0.0000

HISTOGRAMS

NODE	LOWER LIMIT	CELL WIDTH	FREQUENCIES										
7	2.00	1.00	2	3	7	10	14	18	13	25	17	23	39
			27	24	26	25	18	19	14	6	12	8	5
			12	5	8	7	0	3	1	0	0	9	
5	2.00	1.00	81	38	25	37	22	18	15	18	11	7	10
			6	6	4	4	4	2	1	3	2	0	1
			0	0	0	0	0	0	0	0	0	2	

Figure 31. Final Results from GERT Simulation Program for Project 14

from node 2 to node 5 is not taken, we also have an exponential; if the activity from node 2 to node 5 and the activity from node 3 to node 5 are taken, we again have an exponential time to realize node 5 since we have a minimum of two exponentials. The probabilities associated with these events were illustrated earlier. The variability in the time to realize node 7 can be seen from the histogram associated with node 7.

Project 15: Starting of Project Phases Based on Progress to Date

The network in Figure 32 represents a situation in which follow-on

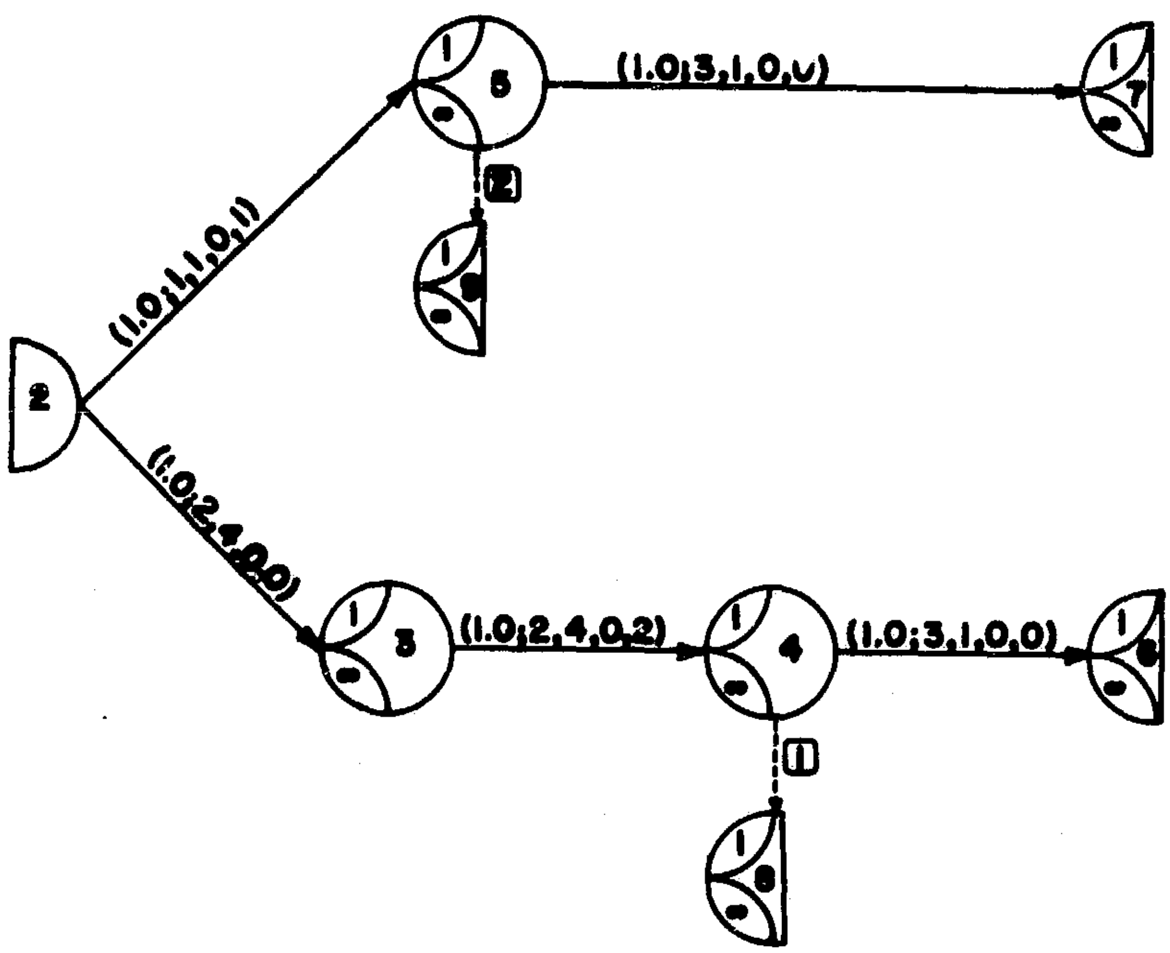


Figure 32. The GERT Network for Project 15

activities are started at a specified point in time even though the current activities have not been complete. The activities of phase I of a project are represented by the branch from node 2 to node 3 and phase II by the branch from node 3 to node 4. Phase III of the project would normally start at node 6. However, if phase II is not completed by 30 time units, management has decided to begin phase III and that an increased effort will be required for phase III. The source node for the network representing the increased effort for phase III is node 7. Since only the concepts involved in using GERTS II are being illustrated, phases I and II are modeled by single branches. In actuality, the branches from node 2 to node 3, and node 3 to node 4 would be represented by complete GERT networks. The branch from node 2 to node 5 represents a clock time. The activity from node 2 to node 5 is labeled activity 1. When activity 1 is realized, it causes node 4 of the network to be replaced by node 8. (Indicated as file 8 on the GERT simulation network modification section.) Node 8 is a terminal node as it has no branches emanating from it. It is not a sink node since the simulation does not stop nor can it stop when this node is reached. It is used as a means for halting the progress of activities in that section of the network. When node 4 is realized and its realization comes after the realization of activity 1, GERTS II transfers to node 8 and seeing no activities emanating from node 8 returns to determine the next activity to be realized. In this manner when time reaches 30 time units node 5 causes node 7 to be activated, and activity 1 causes activities following node 4 to be deleted. If activity 2, the activity between nodes 3 and 4, is realized before activity 1, then node 9 replaces node 5 and the clock does not proceed any further. Then node

6 is the starting node for phase III. There are many other interpretations that can be given to this simple network, and the concepts involved can be useful in many program planning situations.

The data input for the network of Figure 32 is shown in Figure 33, and the GERTS II reports are given in Figures 34, 35 and 36. In this example, the times associated with the activities involved four distribution types. There is no restriction regarding the combining of distribution types on a network.

Project 16: Multiple Modification of a Node During One Realization of a Network

This project illustrates how during the realization of a network a node can be replaced by another node and changed back again. The network for project 16 is shown in Figure 37. It is desired to formulate the network so that the activities emanating from node 5 are dependent on whether activity 1 preceded node 5 or activity 2 preceded node 5. When activity 1 was the last of the two activities performed prior to the realization of node 5, the activities emanating from node 5 should be specified by the branches from node 5 to node 3 and from node 5 to node 7. However, if activity 2 was performed prior to the realization of node 5 and no activity 1 was realized in the interim, then node 5 should be replaced by node 10 with the branches from node 10 to node 3 and from node 10 to node 7 inserted into the network. A similar analysis holds for node 6 except that if neither activities 1 or 2 are performed, then the activities emanating from node 6 are unaltered. An example of a possible sequence may help clarify the above description. A possible

PRITSKER 15 4 25 1969 400 5 20 1267	1	DA15 10
9 1 2 1 2 0 2	2	DA15 20
2 6 7	3	DA15 30
0 1 1 1 1 1 1 1	4	DA15 40
1 1 1 1 1 1 1 1	5	DA15 50
20.0 30.0	6	DA15 60
1.0 1.0	7	DA15 70
30.0	8	DA15 80
8.0 0.0 100.0 1.0		DA15 90
0.0		DA15 100
20.0 0.0 100.0 2.0		DA15 110
1.0 0.0 100.0 2.0		DA15 120
1.0 2 5 1 1 0 1	9	DA15 130
1.0 2 3 4 2 0		DA15 140
1.0 5 7 3 1 0		DA15 150
1.0 3 4 2 4 0 2		DA15 160
1.0 4 6 5 4 0		DA15 170
0		DA15 180
1 4 8 0	10	DA15 190
2 5 9 0		DA15 200
0		DA15 210

Figure 33. Listing of Input Data for Project 15

GERT SIMULATION PROJECT 15 BY PRITSKER
DATE 4/ 25/ 1969

NETWORK DESCRIPTION

START NODE	END NODE	PARAMETER NUMBER	DISTRIBUTION TYPE	COUNT TYPE	ACTIVITY NUMBER	PROBABILITY
2	5	1	1	0	1	1.0000
2	3	4	2	0	0	1.0000
3	4	2	4	0	2	1.0000
4	6	5	4	0	0	1.0000
5	7	3	1	0	0	1.0000

ACTIVITY PARAMETERS

PARAMETER NUMBER	PARAMETERS 1	2	3	4
1	30.0000	0.0000	0.0000	0.0000
2	8.0000	0.0000	100.0000	1.0000
3	0.0000	0.0000	0.0000	0.0000
4	20.0000	0.0000	100.0000	2.0000
5	1.0000	0.0000	100.0000	2.0000

NETWORK MODIFICATIONS

ACTIVITY	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE
1	4	8																		
2	5	9																		

Figure 34. Network Description, Activity Parameters, and Network Modifications for Project 15

```
HIGHEST NODE NUMBER IS     9
NUMBER OF SOURCE NODES IS     1
NUMBER OF SINK NODES IS     2
NUMBER OF NODES TO REALIZE THE NETWORK IS     1
STATISTICS COLLECTED ON     2 NODES
NUMBER OF PARAMETER SETS IS     5
INITIAL RANDOM NUMBER IS 1267     0.0000

SOURCE NODE NUMBERS
      2

SINK NODE NUMBERS
      6    7

STATISTICS COLLECTED ALSO ON NODES
```

NODE	NUMBER RELEASES	NODE TYPE
2	0	1
3	1	1
4	1	1
5	1	1
6	1	1
7	1	1
8	1	1
9	1	1

Figure 35. Significant Network Characteristics for Project 15

GERT SIMULATION PROJECT 15 BY PRITSKER
DATE 4/ 25/ 1969

FINAL RESULTS FOR 400 SIMULATIONS

NODE	PROB./COUNT	MEAN	STD.DEV.	MIN.	MAX.
6	0.7550	25.9183	3.4459	15.7935	37.1201
7	0.2450	30.0000	0.0000	30.0000	30.0000

HISTOGRAMS

NODE	LOWER LIMIT	CELL WIDTH	FREQUENCIES										
6	20.00	1.00	9	10	14	33	32	26	32	35	31	18	23
			21	7	5	4	0	0	1	1	0	0	0
			0	0	0	0	0	0	0	0	0	0	
7	30.00	1.00	0	98	0	0	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	0
			0	0	0	0	0	0	0	0	0	0	

Figure 36. Final Results from GERT Simulation Program for Project 15

sequence of node realizations for this network is 9, 3, 6, 4, 6, 4, 10, 3, 5, 3, 11, 8. Note that node 11 is used as the next to last node since activity 1 preceded the realization of node 6 without activity 2 occurring in the interim. The GERT simulation program is written to allow as many node modifications as required in the realization of the network. The network modifications permit the incorporation of memory or path dependence into the transition probabilities associated with the nodes (states) of the network. The data input for project 16 is given in Fig. 38 with the outputs from GERTS II given in Figures 39, 40 and 41.

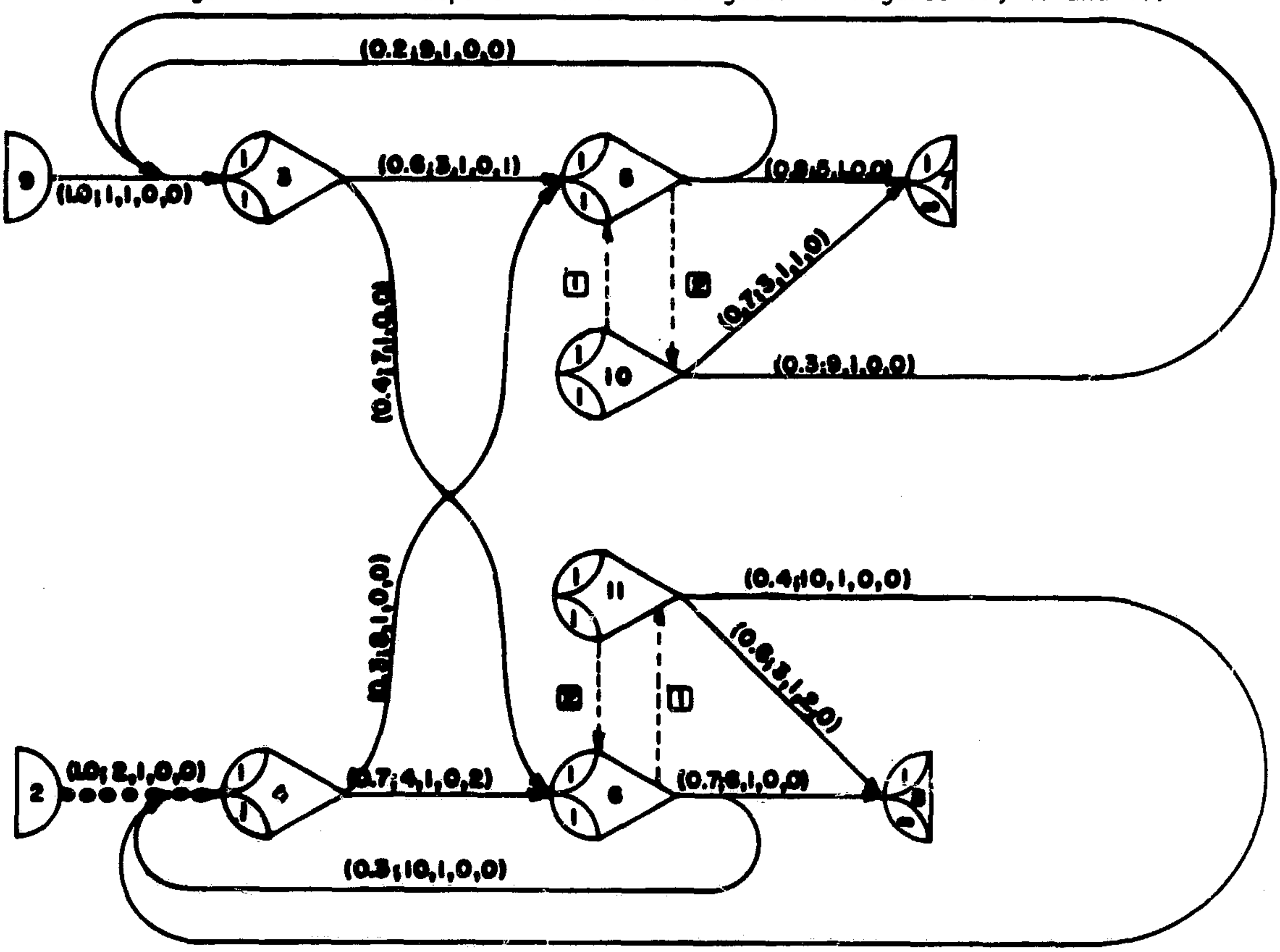


Figure 37. The GERT Network for Project 16

PRITSKER 16 04 22 1969 0400 10 30 1267	1	DA16 10
11 1 2 1 2 2 2	2	DA16 20
9 7 8	3	DA16 30
1 1 1 1 1 1000 1 1	4	DA16 40
1 4 4 4 4 1 1 1 4 4	5	DA16 50
2.0 4.0	6	DA16 60
4.0 2.0	7	DA16 70
1.0	8	DA16 80
1.0		DA16 90
2.0		DA16 100
2.0		DA16 110
0.0		DA16 120
0.0		DA16 130
4.0		DA16 140
5.0		DA16 150
7.0		DA16 160
6.0		DA16 170
1.0 9 3 1 1	9	DA16 180
1.0 2 4 2 1		DA16 190
0.6 3 5 3 1 0 1		DA16 200
0.40 3 6 7 1		DA16 210
0.3 4 5 8 1		DA16 220
0.70 4 6 4 1 0 2		DA16 230
0.8 5 7 5 1		DA16 240
1.0 5 3 9 1		DA16 250
0.20 5 3 9 1		DA16 250
0.7 6 8 6 1		DA16 260
0.30 6 4 10 1		DA16 270
0.7 10 7 3 1 1		DA16 280
0.30 10 3 9 1		DA16 290
0.6 11 8 3 1 2		DA16 300
0.40 11 4 0 1		DA16 310
0		DA16 320
1 6 11 5 5 0	10	DA16 330
2 5 10 6 6 0		DA16 340
0		DA16 350

Figure 38. Listing of Input Data for Project 16

GERT SIMULATION PROJECT 16 BY PRITSKER
DATE 4/ 22/ 1969

NETWORK DESCRIPTION

START NODE	END NODE	PARAMETER NUMBER	DISTRIBUTION TYPE	COUNT TYPE	ACTIVITY NUMBER	PROBABILITY
2	4	2	1	0	0	1.0000
3	5	3	1	0	1	0.6000
3	6	7	1	0	0	0.4000
4	6	4	1	0	2	0.7000
4	5	8	1	0	0	0.3000
5	3	9	1	0	0	1.0000
5	7	5	1	0	0	0.8000
5	3	9	1	0	0	0.2000
6	8	6	1	0	0	0.7000
6	4	10	1	0	0	0.3000
9	3	1	1	0	0	1.0000
10	7	3	1	1	0	0.7000
10	3	9	1	0	0	0.3000
11	8	3	1	2	0	0.6000
11	4	10	1	0	0	0.4000

ACTIVITY PARAMETERS

PARAMETER NUMBER	PARAMETERS 1	2	3	4
1	1.0000	0.0000	0.0000	0.0000
2	1.0000	0.0000	0.0000	0.0000
3	2.0000	0.0000	0.0000	0.0000
4	2.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000
7	4.0000	0.0000	0.0000	0.0000
8	5.0000	0.0000	0.0000	0.0000
9	7.0000	0.0000	0.3000	0.0000
10	6.0000	0.0000	0.0000	0.0000

NETWORK MODIFICATIONS

ACTIVITY	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE	NODE	FILE
1	6	11	5	5																
2	5	10	6	6																

Figure 39. Network Description, Activity Parameters, and Network Modifications for Project 16

HIGHEST NODE NUMBER IS 11
NUMBER OF SOURCE NODES IS 1
NUMBER OF SINK NODES IS 2
NUMBER OF NODES TO REALIZE THE NETWORK IS 1
STATISTICS COLLECTED ON 2 NODES
NUMBER OF PARAMETER SETS IS 10
INITIAL RANDOM NUMBER IS 1267 0.0000

SOURCE NODE NUMBERS
9

SINK NODE NUMBERS
7 8

STATISTICS COLLECTED ALSO ON NODES

NODE	NUMBER RELEASES	NODE TYPE
2	0	1
3	1	4
4	1	4
5	1	4
6	1	4
7	1	1
8	1	1
9	0	1
10	1	4
11	1	4

Figure 40. Significant Network Characteristics for Project 16

GERT SIMULATION PROJECT 16 BY PRITSKER
DATE 4/ 22/ 1969

FINAL RESULTS FOR 400 SIMULATIONS

NODE	PROB./COUNT		MEAN	STD.DEV.	MIN.	MAX.
7	0.0250		71.8000	62.1929	35.0000	245.0000
7		1	1.0000	0.0000	1.0000	1.0000
7		2	0.0000	0.0000	0.0000	0.0000
8	0.9750		28.1282	26.3457	5.0000	158.0000
8		1	0.0000	0.0000	0.0000	0.0000
8		2	0.4513	0.4983	0.0000	1.0000

HISTOGRAMS

NODE	LOWER LIMIT	CELL WIDTH	FREQUENCIES										
7	2.00	4.00	0	0	0	0	0	0	0	0	0	1	0
			3	0	2	0	1	1	0	0	0	1	0
			0	0	0	0	0	0	0	0	0	1	
8	4.00	2.00	0	112	0	0	0	21	0	54	0	5	20
			32	2	0	14	0	23	0	12	9	13	1
			6	4	0	11	0	3	2	4	3	39	

Figure 41. Final Results from GERT Simulation Program for Project 16

CONCLUSIONS AND RECOMMENDATIONS

The following four conclusions can be made with regard to GERTS II.

1. The tailoring of GASP IIA subprograms for specific use within GERTS II has decreased processing time by as much as 50%. The six projects included in this report (400 runs/project) required less than six minutes of GE 425 computer time.
2. GERTS II input requirements are significantly less than the previous GERT simulation program. For the four examples presented in the earlier report the following reductions were made:

Cards Required to Enter Input Network

Example Number	Original GERT Simulation Program	GERTS II
1	25	14
2	46	28
3	55	35
4	86	44

3. The inclusion of new node types, counters, and network modifications makes GERTS II a flexible and useful planning tool.
4. Since GERTS II is written in a basic FORTRAN IV it is easy to modify to meet individual needs.

The following seven recommendations are suggested for research and development on GERT simulation programs.

1. The fundamental need at this time is the extended use of GERTS II to solve problems. One application is given in reference 4. Through such applications, the need for further refinement of the program can be ascertained.

2. The network modifications possible with GERTS II involve the replacement of the node when the node is realized. It is also possible to alter the input side of a node based on the realization of an activity. There are several problems encountered in attempting to do this; such as the determination of the number of releases remaining on the input side of a replaced node. The programming to accomplish this is illustrated below for the situation when the number of releases on the node inserted is set equal to its original number of releases minus the number of realizations of activities that were incident to the node replaced.

```
    5  NEND = NFTBU(NEND)                                 GSPS 260
       NREL(NEND) = NREL(NEND)-1                          GSPS 261
          .
          .
          .
       IF(NREL(NDN)-NRELP(NDN)) 1700,1701,1701            GSPS 351
 1700  NFTTT = NFTBU(NDN)                                 GSPS 352
       NREL(NFTTT) = NREL(NFTTT)-(NRELP(NDN)-NREL(NDN))   GSPS 353
       IF(NREL(NFTTT)) 1702,1702,1701                     GSPS 354
 1702  ATRIB(1) = TNOW                                    GSPS 355
       JTRIB(1) = NFTTT                                   GSPS 356
       CALL FILEM(1,NSET,QSET)                            GSPS 357
 1701  IF(NABA(NPOS+2)) 65,65,68                          GSPS 360
```

3. Study of the statistical characteristics of GERT networks to assist in the design of simulation experiments and the

anaylsis of GERT networks through simulation.

4. For branches with constant times, eliminate the requirement for a parameter set. This can easily be accomplished by using the parameter set number as the constant time. The programming change required is in subroutine SAMPL and is

```
1  DEV = JP                                                   SAMP 120
```

This change was not included due to the problems involved in specifying alternative procedures for incorporating the time parameter.

5. The inclusion of additional distributions from which random samples can be obtained.

6. The automatic calculation of the parameters of the normal distribution which correspond to the parameters of a log-normal distribution. FORTRAN statements required to accomplish this change are shown below.

```
      IF(JTRIB(3)-5) 300,323,300                              DATS 861
323   JJ = JTRIB(2)                                           DATS 862
      PARAM(JJ,4) = SQRT(ALOG((PARAM(JJ,4)/PARAM(JJ,1))**2
     1  +1.0))                                                DATS 863
      PARAM(JJ,3) = ALOG(PARAM(JJ,3))                         DATS 864
      PARAM(JJ,2) = ALOG(PARAM(JJ,2))                         DATS 865
      PARAM(JJ,1) = ALOG(PARAM(JJ,1))-1.05*PARAM(JJ,4)*
     1  PARAM(JJ,4)                                           DATS 866
```

The change was not included since it involves changing parameters of a parameter set, and currently it is possible to use the same parameter set for different distributions.

7. Accumulate total activity time and total costs of performing activities as standard statistical collection variables. This was done for a special application [3]. Since many variables can be associated with an activity, further research is needed in order to develop a general procedure.

BIBLIOGRAPHY

1. Drezner, S. M. and A. A. B. Pritsker, Network Analysis of Countdown. The RAND Corporation, RM-4976-NASA, March, 1966.

2. Elmaghraby, Salah E., "An Algebra for the Analysis of Generalized Activity Networks," Management Science, Vol. 10, No. 3, April 1964, pp. 494-514.

3. Enlow, R. A. and A. A. B. Pritsker, Planning R&D Projects Using GERT, NASA/ERC, NASA-12-2035, Arizona State University, June 1969.

4. Eyestone, David L., Consumer Brand Choice: A GERT Network Approach, Unpublished Engineering Report, Arizona State University, June 1969.

5. Naylor, T. H., et al., Computer Simulation Techniques, New York: John Wiley & Sons, Inc., 1966.

6. Pritsker, A. A. B., GERT: Graphical Evaluation and Review Techniques, The RAND Corporation, RM-4973-NASA, April, 1966.

7. ______, User's Manual for GERT Simulation Program, NASA/ERC, NGR 03-001-034, Arizona State University, July 1968.

8. ______, Definitions and Procedures for GERT Simulation Program, NASA/ERC, NGR 03-001-034, Arizona State University, July 1968.

9. ______, and W. W. Happ, "GERT: Graphical Evaluation and Review Technique, Part I Fundamentals," The Journal of Industrial Engineering, Vol. XVII, No. 5, pp. 267, May, 1966.

10. ______, and P. J. Kiviat, Simulation with GASP II, Englewood Cliffs, New Jersey, Prentice-Hall, Inc., 1969.

11. ______, and G. E. Whitehouse, "GERT: Graphical Evaluation and Review Technqiue, Part II Probabilistic and Industrial Engineering Applications," Journal of Industrial Engineering, Vol. XVII, No. 6, June, 1966.

12. Settles, F. Stanley, Jr., GERT Network Models of Production Economics, PhD Dissertation, Arizona State University, June 1969.

13. Whitehouse, G. E. and A. A. B. Pritsker, "GERT: Part III - Further Statistical Results: Counters, Renewal Times, and Correlations," AIIE Transactions, Vol. 1, No. 1, March 1969, pp. 45-50.

APPENDIX A

GLOSSARY OF GERT SIMULATION SUBPROGRAMS AND VARIABLES

Subprograms

NAME	FUNCTION
AMAX(ARG1, ARG2)	Sets AMAX to the maximum of ARG1 and ARG2.
AMIN(ARG1, ARG2)	Sets AMIN to the minimum of ARG1 and ARG2.
COLCT(X, N, NSET, QSET)	Sets X_i = X and collects: $\sum_i X_i$ in SUMA(N, 2) $\sum_i X_i^2$ in SUMA(N, 2) Number of observations of type N in SUMA(N, 3) $\min_i (X_i)$ in SUMA(N, 4) $\max_i (x_i)$ in SUMA(N, 5)
DATAN(NSET, QSET)	Data input and initialization routine
DRAND(ISEED)	Generates a random number DRAND using the computer system's random number routine.
ERLNG(J)	Generates a random deviate from an ERLNG distribution with parameters in PARAM(J,I), I = 1, 2, 3, 4.
ERROR(J, NSET, QSET)	Prints error code J, NSET, QSET and File 1 and calls SUMRY. Then calls EXIT.
FILEM(JQ, NSET, QSET)	Stores the vector ATRIB in QSET and JTRIB in NSET of File JQ in their proper order.

FINDN(NVAL, JQ, JATT, NSET, QSET)	Locates FINDN of File JQ that has an attribute value in row JATT of NSET equal to NVAL.
GASP(NSET, QSET)	The supervisor of the GERT simulation program. Selects the next scheduled event and controls simulation from start to end.
HISTO(X1, A, W, N)	X1 is a value to be used to form histogram N. A is the lower limit. W is the cell width.
NPOSN(J, NPSSN)	Generates a random deviate from a Poisson distribution with parameter PARAM (J,I), I = 1, 2, 3, 4.
RLOGN(J)	Generates a random deviate from a log-normal distribution with parameters PARAM(J, I), I = 1, 2, 3, 4.
RMOVE(KCOL, JQ, NSET, QSET)	Removes column KCOL from File JQ and stores attribute of NSET in the vector JTRIB and attributes of QSET in ATRIB.
RNORM(J)	Generates a random deviate from a normal distribution with parameters PARAM(J, I), I = 1, 2, 3, 4.
SUMRY(NSET, QSET	Prints the final GERT summary report.
UNFRM(J)	Generates a random deviate from a uniform distribution between PARAM(J,2) and PARAM(J,3).

Variables

The following is an alphabetized list of the variables used in the GERT simulation program stored in COMMON. These variables must be used by a programmer with great care. All variables listed are in COMMON except the arrays NSET and QSET. Variables are dimensioned as required by the GERT simulation program.

ATRIB(1)	Buffer for attribute value being stored in or retrieved from QSET.
ID	Number of columns in total filing array, limited only by available storage.
IM	Number of attribute rows in NSET (IM=5)
IMM	Number of attribute rows of QSET (IMM=1)
ISEED	The initial random number.
JCELS(NHIST, MXC)	Storage array for histograms. (NHIST $\leq$ 20, MXC $\leq$ 32).
JSEED	A local variable used in subroutine DATAN that reads in the initial random number seed value. JSEED must be positive for TNOW to be set to TBEG.
JTRIB(5)	Buffer for attribute values being stored in or retrieved from NSET.
JSINK(NSKS)	JSINK(J) = 0 if node J has not been realized in this simulation; 1 if statistics have been collected for node J in this run.
KOUNT(NCTS)	KNOUNT(K) is the number of times counter K was indexed in a simulation run.
MFA	Identifies the first column in NSET available for storing an event or entity.
MFE(NN)	MFE(J) is the first entry in File J.

MLE(NN)	MLE(J) is the last entry in File J.
MON	Month number.
MSTOP	An indicator for ending the simulation.
MX	Successor row in array NSET. MX = 6.
MXC	Largest number of cells to be used in any histogram (MXC ≤ 32).
MXX	Predecessor row in array NSET. MXX = 7.
NABA(NN)	Array storing nodes affected by realization of activities.
NAME	User's name or identification.
NCLCT	The number of sets of statistics that can be collected in COLCT (NCLCT ≤ 20).
NCROR	The card reader number.
NCTS	Number of counter types in the simulation (NCTS ≤ 4).
NDAY	Day number.
NFTBU(NN)	NFTBU(J) is the node number to be used when node J is realized.
NHIST	The number of histograms that can be generated by HISTO (NHIST ≤ 20)
NN	The largest node number.
NNM	An indicator. If NNM > 0, nodes are modified.
NOQ	The number of files in filing array. (Equal to NN).
NPD	The number of sink nodes.
NPO(NN)	NPO(J) is cell number in NABA(NN) where nodes affected by activity J are stored.
NPRMS	Number of sets of parameters (NPRMS ≤ 300).

NPRNT	The printer number for the computer being used.
NPROJ	Project number of identification if negative permits tracing.
NQ(NN)	NQ(J) is the current number of entries in File J.
NREL(NN)	NREL(J) is the number of releases for node J.
NRELP(NN)	NRELP(J) is the initial number of releases for node J.
NRUN	The number of the current simulation run.
NRUNS	The number of runs remaining, including the one being processed.
NSET(ID*7)	The integer part of the filing array. (ID limited by available storage).
NSINK(NSKS)	The sink node numbers.
NSKS	The number of nodes on which statistics are collected including sink nodes (NSKS $\leq$ 20)
NSORC(NSRC)	The source node numbers
NSKSR	The number of sink nodes remaining to be realized.
NSKST	The number of sink nodes required for a network to be realized.
NSRC	The number of source nodes.
NTYPE(NN)	NTYPE(J) is the node type for node J.
NYR	Year number.

PARAM(NPRMS, 4)	Array for storing parameter values. (NPRMS ≤ 300)
QSET(ID*1)	The real valued part of the filing array. (ID limited by available storage.)
SUMA(NCLCT,J)	Array for storing statistics generated by COLCT.
TNOW	The current time of a simulation.
TOTIM	Total time all simulation runs.
WIDTH(NSKS)	The width of a histogram cell.
XLOW(NSKS)	The lower limit of cell 2 of a histogram.

APPENDIX B

FORTRAN LISTING OF GERTS II

```
      DIMENSION NSET(7000),QSET(1000)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      COMMON NSINK(20),NSKST,NSRC,XLOW(20),NRELP(999),NREL(999),NN,NSKS,
     1NSKSR,NSORC(20),NTYPE(999),WIDTH(20),TOTIM,NCTS,KOUNT(4),JSINK(20)
      COMMON NPO(999),NFTBU(999),NABA(999)
C
C*****SET EQUIPMENT NUMBERS FOR CARD READER(NCRDR) AND PRINTER (NPRNT).
C
      NCRDR=50
      NPRNT=66
      IMN=999
    7 CALL GASP(NSET,QSET)
      GO TO 7
      END
```

```
      FUNCTION AMAX  (ARG1,ARG2)
      IF (ARG1 - ARG2) 2,1,1
    1 AMAX  = ARG1
      RETURN
    2 AMAX  = ARG2
      RETURN
      END
```

```
      FUNCTION AMIN  (ARG1,ARG2)
      IF (ARG1-ARG2) 1,1,2
    1 AMIN  = ARG1
      RETURN
    2 AMIN  = ARG2
      RETURN
      END
```

```
      SUBROUTINE COLCT (X,N,NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      IF (N) 2,2,1
    2 CALL ERROR(90,NSET,QSET)
    1 IF (N- NCLCT) 3,3,2
    3 SUMA(N,1) = SUMA(N,1)+X
      SUMA(N,2) = SUMA(N,2)+X*X
      SUMA(N,3) = SUMA(N,3)+1.0
      SUMA(N,4) = AMIN (SUMA(N,4),X)
      SUMA(N,5) = AMAX (SUMA(N,5),X)
      RETURN
      END
```

```
      SUBROUTINE DATAN(NSET,QSET)
      DIMENSION NSET(1),QSET(1),NABYA(22)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      COMMON NSINK(20),NSKST,NSRC,XLOW(20),NRELP(999),NREL(999),NN,NSKS,
     1NSKSR,NSORC(20),NTYPE(999),WIDTH(20),TOTIM,NCTS,KOUNT(4),JSINK(20)
      COMMON NPO(999),NFTBU(999),NABA(999)
      NRUN = 1
      READ(NCRDR,101) NAME,NPROJ,MON,NDAY,NYR,NRUNS,NPRMS,ID,JSEED,SEED
  101 FORMAT(6A2,I4,2I2,5I4,F10.4)
      IF(NRUNS) 30,30,5
   30 CALL EXIT
   10 FORMAT(25I3)
   14 FORMAT(8F10.2)
C
C*****READ LARGEST NODE NUMBER, NO. OF SINK NODES, NO. OF SOURCE NODES,
C*****NO. OF NODES NECESSARY TO REALIZE THE NETWORK,NODES MOD.,PRT CDE
C
    5 READ(NCRDR,10) NN,NSRC,NPD,NSKST,NSKS,NCTS,NNM
      NSKSR=NSKST
C
C*****READ SOURCE AND SINK NODE NOS.
C
      READ(NCRDR,10) (NSORC(J),J=1,NSRC), (NSINK(K),K=1,NSKS)
C
C*****READ NO. OF RELEASES FOR EACH NODE. NODE NOS. MUST BE .GE. TO 2.
C
      READ(NCRDR,10)(NRELP(I),I=2,NN)
      DO 11 I=2,NN
   11 NREL(I)=NRELP(I)
C
C*****READ NODE TYPE.  1 AND 3 ARE DETERMINISTIC.  2 AND 4 ARE PROBABILI
C*****3 AND 4 CAN BE REALIZED MORE THAN ONCE.
C
      READ(NCRDR,10)(NTYPE(I),I=2,NN)
C
C*****READ LOWER LIMIT AND CELL WIDTH FOR HISTOGRAMS.
C
      READ(NCRDR,14)(XLOW(K),K=1,NSKS)
      READ (NCRDR,14)(WIDTH(K),K=1,NSKS)
      IF(NCTS) 135,135,125
  125 DO 130 J=1,NCTS
  130 KOUNT(J)=0
  135 DO 140 K=1,NSKS
  140 JSINK(K)=1
      NHIST=NSKS
      NCLCT=NSKS*(NCTS+1)
      IM=5
      NOQ=NN
      MXC=32
      IMM=1
      MSTOP=0
      DO 9 I = 1,NPRMS
      READ (NCRDR,106) (PARAM(I,J),J=1,4)
  106 FORMAT(4F10.4)
    9 CONTINUE
      ISEED=JSEED
```

```
      RNUM=DRAND(ISEED)
      TNOW=0.0
C
C*****INITIALIZE NSET
C
      KOL=7777
      KOF = 8888
      KLE = 9999
      MX  = IM +1
      MXX=IM+2
      MAXQS = ID * IMM
      MAXNS = ID * MXX
      DO 35 J=1,MAXQS
   35 QSET(J)=0.0
      DO 40 J=1,MAXNS
   40 NSET(J)=0
      DO 45 I=1,ID
      INDX = I * MXX
      NSET(INDX - 1) = I + 1
   45 NSET(INDX)=I-1
      NSET(MAXNS - 1) = KOF
      DO 3 K = 1,NOQ
      NQ(K)=0
      MFE(K)=0
    3 MLE(K)=0
      MFA = 1
      JTRIB(1)=0
      JTRIB(2)=0
      JTRIB(3)=0
      JTRIB(4)=0
      ATRIB(1)=0.0
      CALL FILEM(1,NSET,QSET)
      DO 300 JS = 1,ID
C
C*****DATA CARD TYPES NINE
C
      READ(NCRDR,1110) ATRIB(1),JQ,(JTRIB(JK),JK=1,IM)
 1110 FORMAT(F8.3,6I3)
      IF(JQ)23,15,320
   23 CALL ERROR(95,NSET,QSET)
  320 CALL FILEM(JQ,NSET,QSET)
  300 CONTINUE
   15 IF(NCLCT)23,111,116
  116 DO 18 I = 1,NCLCT
      DO 17 J =1,3
   17 SUMA(I,J) = 0.
      SUMA(I,4) = 1.0E20
   18 SUMA(I,5)= -1.0E20
  111 IF(NHIST)23,20,118
  118 DO 380 K = 1,NHIST
      DO 380 L = 1,MXC
  380 JCELS(K,L) = 0
   20 WRITE(NPRNT,21)
      WRITE(NPRNT,22) NPROJ,NAME,MON,NDAY,NYR
      WRITE(NPRNT,24)
      DO 80 JQ=2,NOQ
      PROB = 0.0
      LINE = MFE(JQ)
      NT = NTYPE(JQ)
      IF(LINE)80,80,82
```

```
   82 IB = (LINE-1) * MXX + 1                                             DATS1060
      IE = IB + IM-1                                                      DATS1070
      IQ = (LINE-1) * IMM + 1                                             DATS1080
      DPROB = QSET(IQ)                                                    DATS1090
      GO TO (72,73,72,73,72,73,72,73),NT                                  DATS1100
   73 QSET(IQ)=QSET(IQ)+PROB                                              DATS1110
      PROB = PROB + DPROB                                                 DATS1120
   72 WRITE(NPRNT,12) JQ,(NSET(I),I=IB,IE),DPROB                          DATS1130
      LINE = NSET(IE+1)                                                   DATS1140
      IF(LINE - 7777) 82,80,84                                            DATS1150
   84 WRITE(NPRNT,199)                                                    DATS1160
   80 CONTINUE                                                            DATS1170
      WRITE(NPRNT,26)                                                     DATS1180
      DO 50 I = 1,NPRMS                                                   DATS1190
   50 WRITE(NPRNT,28) I,(PARAM(I,J),J=1,4)                                DATS1200
   21 FORMAT(1H1)                                                         DATS1210
   22 FORMAT(32X,23HGERT SIMULATION PROJECT,I4,4H BY ,6A2/45X,4HDATE,     DATS1220
     1I3,1H/,I3,1H/,I5//)                                                 DATS1230
   24 FORMAT(42X,23H**NETWORK DESCRIPTION**//16X5HSTART,6X3HEND,6X,       DATS1240
     19HPARAMETER,6X,12HDISTRIBUTION,6X,5HCOUNT,6X,8HACTIVITY,6X,         DATS1250
     2 11HPROBABILITY/                                                    DATS1260
     317X4HNODE,6X,4HNODE,7X,6HNUMBER,11X4HTYPE,11X4HTYPE,7X,6HNUMBER/)  DATS1270
   12 FORMAT(I19,I11,I12,I16,2I14,F17.4)                                  DATS1280
  199 FORMAT(///36X26HERROR EXIT, TYPE 70 ERROR.)                         DATS1290
   26 FORMAT(//39X23H**ACTIVITY PARAMETERS**//16X9HPARAMETER32X,10HPARAMDATS1300
     1ETERS/17X6HNUMBER,16X1H1,13X1H2,13X,1H3,13X1H4/)                    DATS1310
   28 FORMAT(I21,7X,4F14.4)                                               DATS1320
      DO 1024 J=1,IMN                                                     DATS1340
 1024 NPO(J)=0                                                            DATS1350
      IF(NNM) 1020,1020,1023                                              DATS1370
 1023 WRITE(NPRNT,1004)                                                   DATS1380
 1004 FORMAT(//39X25H**NETWORK MODIFICATIONS**//12X,8HACTIVITY,10(1X,9HNDATS1390
     1ODE FILE)/)                                                         DATS1400
 1025 DO 1035 NPOS=1,IMN                                                  DATS1410
      READ(NCRDR,1005) NACTN,NABYA                                        DATS1420
 1005 FORMAT(25I3)                                                        DATS1430
      IF(NACTN) 1036,1036,1037                                            DATS1440
 1037 NPO(NACTN)=NPOS                                                     DATS1450
      DO 1038 I=1,22,2                                                    DATS1460
      IF(NABYA(I)) 1040,1040,1039                                         DATS1470
 1039 NABA(NPOS)=NABYA(I)                                                 DATS1480
      NABA(NPOS+1)=NABYA(I+1)                                             DATS1490
 1038 NPOS=NPOS+2                                                         DATS1500
 1040 NABA(NPOS)=0                                                        DATS1510
      IM1=I-1                                                             DATS1530
      WRITE(NPRNT,1006) NACTN,(NABYA(J),J=1,IM1)                          DATS1540
 1006 FORMAT(I17,2X,10(I5,I5))                                            DATS1550
 1035 CONTINUE                                                            DATS1560
 1036 NABA(NPOS)=0                                                        DATS1570
 1020 DO 1041 KK=1,NN                                                     DATS1580
 1041 NFTBU(KK)=KK                                                        DATS1590
      IF(NPROJ) 1591,1592,1592                                            DATS1591
 1591 IMN1=IMN-1                                                          DATS1592
      READ(NCRDR,10) NABA(IMN),NABA(IMN1)                                 DATS1593
      WRITE(NPRNT,21)                                                     DATS1594
 1592 WRITE(NPRNT,221)NN,NSRC,NPD ,NSKST,NSKS,NPRMS,JSEED,SEED            DATS1600
      WRITE(NPRNT,223) (NSORC(J),J=1,NSRC)                                DATS1610
      WRITE(NPRNT,225) (NSINK(K),K=1,NPD)                                 DATS1620
      KPP=NPD+1
      WRITE(NPRNT,227) (NSINK(K),K=KPP,NSKS)                              DATS1630
```

```
      WRITE(NPRNT,229)                                                  DATS1640
      DO 231 J=2,NN                                                     DATS1650
  231 WRITE(NPRNT,233) J,NRELP(J),NTYPE(J)                              DATS1660
  221 FORMAT(1H1,14X22HHIGHEST NODE NUMBER IS,I5,/15X,25HNUMBER OF SOURCDATS1670
     1E NODES IS,I5/15X,23HNUMBER OF SINK NODES IS,I5/15X41HNUMBER OF NODDATS1680
     2DES TO REALIZE THE NETWORK IS,I5/15X,23HSTATISTICS COLLECTED ON,I5DATS1690
     3,6H NODES/15X,27HNUMBER OF PARAMETER SETS IS,I5/15X24HINITIAL RANDDATS1700
     4OM NUMBER IS, I5,F10.4/)                                          DATS1710
  223 FORMAT(15X,19HSOURCE NODE NUMBERS/17X,10(10I5))                   DATS1720
  225 FORMAT(/15X,17HSINK NODE NUMBERS/17X,10(10I5))                    DATS1730
  227 FORMAT(/15X,34HSTATISTICS COLLECTED ALSO ON NODES/17X,10(10I5))   DATS1740
  229 FORMAT(/25X,5HNODE 5X, 6HNUMBER,5X,4HNODE/34X,8HRELEASES,4X,4HTYPEDATS1750
     1/)                                                                DATS1760
  233 FORMAT(I28,I10,I10)                                               DATS1770
      RETURN                                                            DATS1780
      END                                                               DATS1790



      FUNCTION DRAND(ISEED)                                             DRDS  10
      DRAND=RANDF(ISEED)                                                DRDS  20
      RETURN                                                            DRDS  30
      END                                                               DRDS  40



      FUNCTION ERLNG (J)                                                ERLG  10
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,ERLG  20
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,                 ELNG  30
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32) ERLG  40
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)         ERLG  50
      K = PARAM(J,4)                                                    ERLG  60
      IF(K-1) 8,10,10                                                   ERLG  70
    8 WRITE(NPRNT,20) J                                                 ERLG  80
   20 FORMAT(/16HK = 0 FOR ERLNG,I7)                                    ERLG  90
      CALL EXIT                                                         ERLG 100
   10 ERLNG = 0.                                                        ERLG 110
      DO 2 I = 1,K                                                      ERLG 120
      RNUM=DRAND(ISEED)                                                 ERLG 130
    2 ERLNG = ERLNG - PARAM(J,1)*ALOG(RNUM)                             ERLG 140
      IF(ERLNG-PARAM(J,2))7,5,6                                         ERLG 150
    7 ERLNG  = PARAM (J,2)                                              ERLG 160
    5 RETURN                                                            ERLG 170
    6 IF(ERLNG  - PARAM (J,3))5,5,4                                     ERLG 180
    4 ERLNG  = PARAM (J,3)                                              ERLG 190
      RETURN                                                            ERLG 200
      END                                                               ERLG 210
```

```
      SUBROUTINE ERROR(J,NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      WRITE(NPRNT,99)
      WRITE (NPRNT,100) J,TNOW
  100 FORMAT(//36X16HERROR EXIT, TYPE,I3,7H ERROR.//21H FILE STATUS AT
     1TIME,F10.4/)
      WRITE (NPRNT,200)
  200 FORMAT(30X,4HNSET/)
      DO 210 I=1,ID
      IL=(I-1)*MXX+1
      IV=IL+MXX-1
  210 WRITE(NPRNT,90) I,(NSET(IJ),IJ=IL,IV)
   90 FORMAT(13X,I5,5X,12I8)
      WRITE (NPRNT,202)
  202 FORMAT(//30X,4HQSET/)
      DO 215 I=1,ID
      IL=(I-1)*IMM+1
      IV=IL+IMM-1
  215 WRITE(NPRNT,95) I,(QSET(IJ),IJ=IL,IV)
   95 FORMAT(13X,I5,4X,10(E12.6,2X))
      WRITE (NPRNT,99)
   99 FORMAT(1H1)
      IF(NCLCT)9,9,8
    8 WRITE (NPRNT,98)
   98 FORMAT(/11H ARRAY SUMA,/)
      DO 110 I=1,NCLCT
  110 WRITE (NPRNT,80) I,(SUMA(I,K),K=1,5)
   80 FORMAT(I10,5F10.4)
      WRITE (NPRNT,99)
    9 IF(NHIST) 11,11,12
   12 WRITE (NPRNT,96)
   96 FORMAT (/12H ARRAY JCELS/)
      NCL=MXC
      DO 112 I=1,NHIST
  112 WRITE (NPRNT,26) I,(JCELS(I,K),K=1,NCL)
   26 FORMAT(17X,I3,5X,11I4/(25X,11I4))
   11 NFOOL = 0
      IF (NFOOL) 3,4,3
    3 RETURN
    4 CALL EXIT
      END
```

```
      SUBROUTINE FILEM (JQ,NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
C
C*****TEST TO SEE IF THERE IS AN AVAILABLE COLUMN FOR STORAGE
C
      IF (MFA - ID ) 2,2,3
    3 WRITE (NPRNT,4)
    4 FORMAT (//24H OVERLAP SET GIVEN BELOW/)
      CALL ERROR (87,NSET,QSET)
C
C*****PUT ATTRIBUTE VALUES IN FILE
C
    2 INDX = (MFA - 1) * IMM
      KOL = 7777
      KOF = 8888
      KLE = 9999
      DO 1 I = 1,IMM
      INDX = INDX + 1
    1 QSET(INDX) = ATRIB(I)
      INDX = (MFA - 1) * MXX
      DO 10 I = 1,IM
      INDX = INDX + 1
   10 NSET(INDX) = JTRIB(I)
      MFEX=MFE(JQ)
      KNT = 2
      KS=1
      INDX=MFA*MXX-1
      NXFA = NSET(INDX)
      MLEX=MLE(JQ)
      IF(MLEX)100,27,11
  100 CALL ERROR (88,NSET,QSET)
   27 INDX=MFA*MXX
      NSET(INDX) = KLE
      MFE(JQ) = MFA
   17 INDX = MFA * MXX - 1
      NSET(INDX) = KOL
      MLE(JQ) = MFA
   14 MFA =NXFA
      IF(MFA-KOF) 237,238,238
  237 INDX=NXFA*MXX
      NSET(INDX) = KLE
  238 NQ(JQ)=NQ(JQ)+1
      RETURN
   11 INDX1=(MFA-1)*IMM+KS
      INDX2 = (MLEX - 1) * IMM + KS
      IF(JQ-1) 241,241,242
  241 IF (QSET(INDX1) - QSET(INDX2)) 12,13,13
  242 IF (QSET(INDX1) - QSET(INDX2)) 13,13,12
   13 INDX = MLEX * MXX - 1
      MSU = NSET(INDX)
      NSET(INDX) = MFA
      INDX = MFA * MXX
      NSET(INDX) = MLEX
      GO TO (18,17),KNT
   18 INDX = MFA * MXX - 1
```

```
      NSET(INDX) = MSU
      INDX = MSU * MXX
      NSET(INDX) = MFA
      GO TO 14
   12 KNT = 1
      INDX = MLEX * MXX
      MLEX = NSET(INDX)
      IF(MLEX-KLE)  11,16,11
   16 INDX = MFA * MXX
      NSET(INDX) = KLE
      MFE(JQ) = MFA
      INDX = MFA * MXX - 1
      NSET(INDX) = MFEX
      INDX = MFEX * MXX
      NSET(INDX) = MFA
      GO TO 14
      END
```

```
      SUBROUTINE GASP(NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      COMMON NSINK(20),NSKST,NSRC,XLOW(20),NRELP(999),NREL(999),NN,NSKS,
     1NSKSR,NSORC(20),NTYPE(999),WIDTH(20),TOTIM,NCTS,KOUNT(4),JSINK(20)
      COMMON NPO(999),NFTBU(999),NABA(999)
      CALL DATAN(NSET,QSET)
C
C*****OBTAIN NEXT EVENT WHICH IS FIRST ENTRY IN FILE 1.  ATRIB(1) IS
C*****EVENT TIME, JTRIB(1) IS EVENT CODE
C
   11 CALL RMOVE(MFE(1),1,NSET,QSET)
      TNOW=ATRIB(1)
      NEND=JTRIB(1)
C
C*****IF END NODE IS ZERO START NETWORK BY SCHEDULING ACTIVITIES
C*****FROM EACH SOURCE NODE
C
      IF(NEND)4,4,5
    4 DO 10 N=1,NSRC
      M=NSORC(N)
      CALL SCHAT(M,NSET,QSET)
   10 CONTINUE
      GO TO 42
C
C*****REDUCE NO. OF RELEASES FOR END NODE BY 1.
C*****TEST TO SEE IF NODE IS RELEASED.
C
    5 NREL(NEND)=NREL(NEND)-1
      IF(NPROJ) 511,512,512
  511 IF(NRUN-NABA(IMN)) 512,514,514
  514 IMN1=IMN-1
      IF(NRUN-NABA(IMN1)) 515,515,512
  515 WRITE(NPRNT,598) TNOW,JTRIB
  598 FORMAT(/5X,7HAT TIME,F8.2,17H ACTIVITY ON NODE,I5,16H WITH ATTRIBU
     1TES,4I5,13H WAS REALIZED)
  512 IF(JTRIB(4)) 665,665,60
   60 L=JTRIB(4)
      KOUNT(L)=KOUNT(L)+1
  665 IF(JTRIB(5)) 65,65,66
   66 JJJ=JTRIB(5)
      IF(NPO(JJJ)) 65,65,67
   67 NPOS=NPO(JJJ)
   69 NDN=NABA(NPOS)
      NFTBU(NDN)=NABA(NPOS+1)
      IF(NABA(NPOS+2)) 65,65,68
   68 NPOS=NPOS +2
      GO TO 69
   65 IF(NREL(NEND))9,7,42
C
C*****TEST TO SEE IF NODE CAN BE RELEASED MORE THAN ONCE.
C
    9 IF(NTYPE(NEND)-2)42,42,50
C
C*****TEST TO SEE IF STATISTICS ARE TO BE COLLECTED ON THE NODE.
C
```

```
      7 DO 8 K=1,NSKS                                                   GSPS 430
        IF(NEND-NSINK(K))8,39,8                                         GSPS 440
     39 IF(JSINK(K))50,50,40                                            GSPS 450
      8 CONTINUE                                                        GSPS 460
     50 NT=NTYPE(NEND)                                                  GSPS 470
        GO TO (52,52,52,52,52,52,54,54),NT                              GSPS 471
     54 KCOL=NFIND(NEND,1,1,NSET,QSET)                                  GSPS 472
        IF(KCOL) 52,52,56                                               GSPS 473
     56 CALL RMOVE(KCOL,1,NSET,QSET)                                    GSPS 474
        GO TO 54                                                        GSPS 475
     52 CALL SCHAT(NEND,NSET,QSET)                                      GSPS 476
        GO TO (42,42,42,42,51,51,51,51),NT                              GSPS 480
     51 NREL(NEND)=NRELP(NEND)                                          GSPS 490
        GO TO 42                                                        GSPS 500
C
C*****REDUCE NO. OF SINK NODES REALIZED BY ONE.  COLLECT STATISTICS.    GSPS 510
C
     40 IF(K-NPD) 440,440,441                                           GSPS 515
    440 NSKSR=NSKSR-1                                                   GSPS 520
    441 JSINK(K)=0                                                      GSPS 530
        CALL HISTO(TNOW,XLOW(K),WIDTH(K),K)                             GSPS 540
        NL=K+(K-1)*NCTS                                                 GSPS 550
        CALL COLCT(TNOW,NL,NSET,QSET)                                   GSPS 560
        IF(NCTS) 45,45,46                                               GSPS 570
     46 DO 70 I=1,NCTS                                                  GSPS 580
        NL=NL+1                                                         GSPS 590
        XC=KOUNT(I)                                                     GSPS 600
     70 CALL COLCT(XC,NL,NSET,QSET)                                     GSPS 610
C
C*****TEST TO SEE IF NETWORK SIMULATION IS COMPLETE.                    GSPS 620
C
     45 IF(NSKSR) 22,100,999                                            GSPS 630
    999 IF(NQ(NEND))22,42,50                                            GSPS 640
     22 CALL ERROR(22,NSET,QSET)                                        GSPS 650
C
C*****NETWORK HAS BEEN SIMULATED ONE MORE TIME.  ADD TNOW TO TOTAL TIME.GSPS 660
C
    100 TOTIM=TOTIM+TNOW                                                GSPS 670
C
C*****TEST TO SEE IF ALL RUNS HAVE BEEN MADE.                           GSPS 680
C
        IF(NRUNS-NRUN)110,110,115                                       GSPS 690
C
C*****REINITIALE FOR ANOTHER RUN BY REMOVING ALL EVENTS FROM EVENT      GSPS 700
C*****FILE AND RESETTING NETWORK VALUES.                                GSPS 710
C
    115 IF(NQ(1))22,120,116                                             GSPS 720
    116 CALL RMOVE(MFE(1),1,NSET,QSET)                                  GSPS 730
        GO TO 115                                                       GSPS 740
    120 NRUN=NRUN+1                                                     GSPS 750
        JTRIB(1)=0                                                      GSPS 760
        ATRIB(1)=0.                                                     GSPS 770
        CALL FILEM(1,NSET,QSET)                                         GSPS 780
        TNOW=0.                                                         GSPS 790
        NSKSR=NSKST                                                     GSPS 800
        DO 80 K=1,NSKS                                                  GSPS 810
     80 JSINK(K)=1                                                      GSPS 820
        DO 85 J=1,NCTS                                                  GSPS 830
     85 KOUNT(J)=0                                                      GSPS 840
        DO 37 IJ=1,NN                                                   GSPS 850
```

```
      NFTBU(IJ)=IJ
   37 NREL(IJ)=NRELP(IJ)
      GO TO 42
  110 MSTOP=-1
      NRUNS=1
      TNOW=TOTIM
C
C*****IS IT TIME TO STOP
C
   42 IF(MSTOP)44,11,11
   44 CALL SUMRY(NSET,QSET)
      RETURN
      END
```

```
      SUBROUTINE HISTO (X1,A,W,N)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      IF (N-NHIST) 11,11,2
    2 WRITE(NPRNT,250) N
  250 FORMAT(19H ERROR IN HISTOGRAM,I4//)
      CALL EXIT
   11 IF(N)2,2,3
C
C*****TRANSLATE X1 BY SUBTRACTING A IF X.LE.A THEN ADD 1 TO FIRST CELL
C
    3 X = X1 - A
      IF (X)6,7,7
    6 IC = 1
      GO TO 8
C
C*****DETERMINE CELL NUMBER IC. ADD 1 FOR LOWER LIMIT CELL AND 1 FOR
C*****TRUNCATION
C
    7 IC = X/W + 2.  +.0001
      IF(IC - MXC     )8,8,9
    9 IC = MXC
    8 JCELS(N,IC) = JCELS(N,IC) + 1
      RETURN
      END
```

```
      FUNCTION NFIND(NVAL,JQ,JATT,NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(200),MLE(200),NQ(200),PARAM(300,4),SUMA(100,5)
      NEXTK=MFE(JQ)
      IF(NEXTK) 16,1,20
   16 CALL ERROR(89,NSET,QSET)
    1 NFIND=0
      RETURN
   20 INDX=(NEXTK-1)*MXX +JATT
      IF(NSET(INDX)-NVAL) 4,21,4
    4 INDS=NEXTK*MXX -1
      NEXTK=NSET(INDS)
      IF(NEXTK-7777) 20,1,1
   21 NFIND=NEXTK
      RETURN
      END
```

```
      SUBROUTINE NPOSN(J,NPSSN)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      NPSSN = 0
      P = PARAM (J,1)
      IF (P-6.0) 2,2,4
    2 Y = EXP (-P)
      X = 1.0
    3 RNUM=DRAND(ISEED)
      X=X*RNUM
      IF (X-Y) 6,8,8
    8 NPSSN = NPSSN+1
      GO TO 3
    4 TEMP=PARAM (J,4)
      PARAM(J,4) = (PARAM(J,1))**.5
      NPSSN=RNORM(J) +.5
      PARAM (J,4)=TEMP
      IF(NPSSN)4,6,6
    6 KK=PARAM (J,2)
      KKK=PARAM (J,3)
      NPSSN=KK+NPSSN
      IF(NPSSN-KKK)7,7,9
    9 NPSSN = PARAM (J,3)
    7 RETURN
      END
```

```
      FUNCTION RLOGN (J)
C
C*****THE PARAMETERS USED WITH RLOGN ARE THE MEAN AND STANDARD DEVIATION
C*****OF A NORMAL DISTRIBUTION
C
      VA= RNORM (J)
      RLOGN=EXP(VA)
      RETURN
      END
```

```
      SUBROUTINE RMOVE  (KCOLL,JQ,NSET,QSET)
      DIMENSION NSET(1),QSET(1),KCOLL(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
C
C*****THE DUMMY ARRAY KCOLL IS USED AS AN ARGUMENT TO FORCE THE CALL
C*****     BY NAME OPTION ON COMPUTERS SUCH AS THE IBM 360.  THE CALL
C*****     BY NAME OPTION IS REQUIRED FOR PROPER OPERATION OF THIS
C*****     SUBROUTINE.
C
      KOL = 7777
      KOF = 8888
      KLE = 9999
      KCOL = KCOLL(1)
      IF (KCOL) 16,16,2
   16 CALL ERROR (97,NSET,QSET)
C
C*****PUT VALUES OF KCOL IN ATTRIB
C
    2 INDX=(KCOL-1)*IMM
      DO 3 I = 1,IMM
      INDX = INDX + 1
    3 ATRIB(I) = QSET(INDX)
      INDX = (KCOL - 1) * MXX
      DO 10 I =1,IM
      INDX = INDX + 1
   10 JTRIB(I) = NSET(INDX)
      INDX=(KCOL-1)*IMM
      DO 32 I=1,IMM
      INDX = INDX + 1
   32 QSET(INDX) = 0.0
      INDX=(KCOL-1)*MXX
      DO 1300 I= 1,IM
      INDX = INDX + 1
 1300 NSET(INDX) = 0
      INDX=KCOL*MXX
      JL = NSET(INDX - 1)
      JK = NSET(INDX)
      IF (JL- KOL) 33,34,33
   33 IF (JK- KLE) 35,36,35
   35 INDX = JK * MXX - 1
      NSET(INDX) = JL
      INDX = JL * MXX
      NSET(INDX) = JK
   37 INDX=KCOL*MXX-1
      NSET(INDX) = MFA
      NSET(INDX+1) = KLE
      IF(MFA-KOF) 234,235,235
  234 INDX=MFA*MXX
      NSET(INDX)=KCOL
  235 MFA=KCOL
      NQ(JQ) = NQ(JQ)-1
      RETURN
   36 INDX = JL * MXX
      NSET(INDX) = KLE
      MFE(JQ) = JL
      GO TO 37
```

```
34 IF (JK-KLE)  38,39,38
38 INDX = JK *MXX - 1
   NSET(INDX) = KOL
   MLE(JQ) = JK
   GO TO 37
39 MFE(JQ) = 0
   MLE(JQ) = 0
   GO TO 37
   END
```

```
      FUNCTION RNORM (J)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      RA=DRAND(ISEED)
      RB=DRAND(ISEED)
      V=(-2.0*ALOG(RA))**0.5*COS (6.283*RB)
      RNORM  = V*PARAM (J,4) + PARAM (J,1)
      IF (RNORM -PARAM (J,2)) 6,7,8
    6 RNORM  = PARAM (J,2)
    7 RETURN
    8 IF (RNORM -PARAM (J,3)) 7,7,9
    9 RNORM  = PARAM (J,3)
      RETURN
      END
```

```
      FUNCTION UNFRM (A,B)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      RNUM=DRAND(ISEED)
      UNFRM  = A+(B-A)*RNUM
      RETURN
      END
```

```
      SUBROUTINE SAMPL(DEV,NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      COMMON NSINK(20),NSKST,NSRC,XLOW(20),NRELP(999),NREL(999),NN,NSKS,
     1NSKSR,NSORC(20),NTYPE(999),WIDTH(20),TOTIM,NCTS,KOUNT(4),JSINK(20)
      JP=JTRIB(2)
      JD=JTRIB(3)
      GO TO (1,2,3,4,5,6),JD
    1 DEV=PARAM(JP,1)
      RETURN
    2 DEV=RNORM(JP)
      RETURN
    3 A=PARAM(JP,2)
      B=PARAM(JP,3)
      DEV=UNFRM(A,B)
      RETURN
    4 DEV=ERLNG(JP)
      RETURN
    5 DEV=RLOGN(J)
      RETURN
    6 CALL NPOSN(J,NPSSN)
      DEV=NPSSN
      RETURN
      END
```

```
      SUBROUTINE SCHAT(NODE,NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      COMMON NSINK(20),NSKST,NSRC,XLOW(20),NRELP(999),NREL(999),NN,NSKS,
     1NSKSR,NSORC(20),NTYPE(999),WIDTH(20),TOTIM,NCTS,KOUNT(4),JSINK(20)
      COMMON NPO(999),NFTBU(999),NABA(999)
C
C*****NEXT IS LOCATION OF FIRST ENTRY IN FILE OF ACTIVITIES WITH START
C*****NODE- NODE. NT IS THE NODE TYPE.
C
      NODE=NFTBU(NODE)
      NT=NTYPE(NODE)
      NEXT=MFE(NODE)
      IF(NEXT) 4,4,5
C
C*****GENERATE A RANDOM NUMBER RNUM.
C
    5 RNUM=DRAND(ISEED)
C
C*****OBTAIN STARTING LOCATION IN NSET OF ENTRY.
C
    1 INDXN=(NEXT-1)*MXX
C
C*****TEST IF DETERMINISTIC OR PROBABILISTIC NODE.
C
      GO TO (100,200,100,200,100,200,100,200),NT
  100 DO 10 I=1,IM
      INDX=INDXN+I
   10 JTRIB(I)=NSET(INDX)
C
C*****OBTAIN SAMPLE FOR ACTIVITY.
C
      CALL SAMPL(DEV,NSET,QSET)
      ATRIB(1)=TNOW+DEV
C
C*****FILE END OF ACTIVITY EVENT IN EVENT FILE.
C
      CALL FILEM(1,NSET,QSET)
      GO TO (110,4,110,4,110,4,110,4),NT
C
C*****DETERMINE IF OTHER ACTIVITIES ARE IN FILE.
C
  110 INDX=INDXN+MX
      NEXT=NSET(INDX)
      IF(NEXT-7777)1,4,23
   23 CALL ERROR(23,NSET,QSET)
    4 RETURN
  200 INDX=(NEXT-1)*IMM+1
C
C*****TEST RNUM AGAINST PROBABILITY (CUM.) OF REALIZING THE ACTIVITY.
C
      IF(QSET(INDX)-RNUM)110,110,100
      END
```

```
      SUBROUTINE SUMRY (NSET,QSET)
      DIMENSION NSET(1),QSET(1)
      COMMON ID,IM,MFA,MSTOP,MX,MXC,NCLCT,NHIST,NPRNT,NCRDR,NNM,IMN,NPD,
     1NOQ,NPRMS,NRUN,NRUNS,ISEED,TNOW,MXX,IMM,NYR,SEED,
     2MAXQS,MAXNS,ATRIB(1),JTRIB(5),NAME(6),NPROJ,MON,NDAY,JCELS(20,32)
      COMMON MFE(999),MLE(999),NQ(999),PARAM(300,4),SUMA(100,5)
      COMMON NSINK(20),NSKST,NSRC,XLOW(20),NRELP(999),NREL(999),NN,NSKS,
     1NSKSR,NSORC(20),NTYPE(999),WIDTH(20),TOTIM,NCTS,KOUNT(4),JSINK(20)
      COMMON NPO(999),NFTBU(999),NABA(999)
   21 FORMAT(1H1)
   10 FORMAT(32X,23HGERT SIMULATION PROJECT,I4,4H BY ,6A2/45X,4HDATE,I3,
     11H/,I3,1H/,I5//)
   15 FORMAT(//31X19H**FINAL RESULTS FOR,I5,14H SIMULATIONS**/)
   16 FORMAT(16X4HNODE3X11HPROB./COUNT,5X4HMEAN,4X8HSTD.DEV.,4X,
     14HMIN.,6X4HMAX.)
   18 FORMAT(//43X,14H**HISTOGRAMS**//20X        ,5HLOWER,3X,4HCELL/13X
     1,4HNODE,3X,5HLIMIT,3X,5HWIDTH,31X,11HFREQUENCIES )
   19 FORMAT(/I16,F9.2,F8.2,5X,11I6/(38X,11I6))
      XRUNS = NRUN
      WRITE(NPRNT,21)
      WRITE (NPRNT,10) NPROJ,NAME,MON,NDAY,NYR
      WRITE (NPRNT,15) NRUN
      WRITE(NPRNT,16)
      NCTS1=NCTS+1
      DO 6 K=1,NSKS
      DO 6 J=1,NCTS1
      I=(K-1)*NCTS1+J
      JJ=J-1
      IF(SUMA(I,3))5,62,61
    5 WRITE (NPRNT,199)
  199 FORMAT(///36X26HERROR EXIT, TYPE 98 ERROR.)
   62 WRITE(NPRNT,63) NSINK(K),JJ
   63 FORMAT(I18,I13,9X,18HNO VALUES RECORDED)
      GO TO 6
   61 XS=SUMA(I,1)
      XSS=SUMA(I,2)
      XN = SUMA(I,3)
      AVG=XS/XN
      STD=(((XN*XSS)-(XS*XS))/(XN*(XN-1.0)))**0.5
      IF(J-1) 41,41,42
   41 PROB=XN/XRUNS
      WRITE(NPRNT,17) NSINK(K),PROB,AVG,STD,SUMA(I,4),SUMA(I,5)
   17 FORMAT(/I18,F9.4,7X,4F10.4)
      GO TO 6
   42 WRITE(NPRNT,20) NSINK(K),JJ,AVG,STD,SUMA(I,4),SUMA(I,5)
   20 FORMAT(I18,I13,3X,4F10.4)
    6 CONTINUE
      WRITE(NPRNT,18)
      DO 82 I=1,NHIST
      NCL=MXC
      WRITE(NPRNT, 19) NSINK(I),XLOW(I),WIDTH(I),(JCELS(I,J),J=1,NCL)
   82 CONTINUE
      WRITE(NPRNT,21)
      RETURN
      END
```

APPENDIX C

DESCRIPTION OF DATA INPUT FOR GERTS II

DATA CARD 1

Field 1	The analyst's name (6A2,1)
Field 2	The project number (I4,1) (If negative, data card 11 is required to indicate the runs to be traced).
Field 3	The month number (I2,1)
Field 4	The day number (I2,1)
Field 5	The year (I4,1)
Field 6	The number of times the network is to be simulated (I4,1)
Field 7	The number of activities with different time characteristics (I4,1)
Field 8	The number of branches in the network plus the number of activities which can occur simultaneously. The maximum number of these which can be accomodated on the GE 425 is 1000 (I4,1)
Field 9	An integer random number seed (I4,1)
Field 10	A floating point random number seed (F10.4,1)

DATA CARD 2

Field 1	The largest node number of the network including all possible modifications to the network (I3,1). The smallest node number permitted is 2. The largest node number permitted on the GE 425 is 999.
Field 2	Number of source nodes (I3,1). A maximum of 20 is permitted on the GE 425 version.
Field 3	Number of sink nodes (I3,1). The maximum value for the GE 425 version is 20.
Field 4	Number of sink nodes that must be realized before the network is realized (I3,1). A maximum of 20 is permitted in the GE 425 version.

Field 5 — Number of nodes which statistics are to be collected on, including all sink nodes (I3,1). A maximum of 20 is permitted.

Field 6 — Number of types of counts (I3,1). A maximum of 4 is permitted.

Field 7 — A 1 if network modifications exist; a 0 otherwise (I3,1)

DATA CARD 3

Field 1 — The node numbers for all source nodes (I3, the number of source nodes). For example, if there are two source nodes (a 2 in Field 2 of Data Card 2) and if the source node numbers are 4 and 6, then 004006 would appear in columns 1 through 6 of Data Card 3.

Field 2 — The node numbers for all sink nodes (I3, the number of sink nodes).

Field 3 — The node numbers for the nodes which are not sink nodes on which statistics are to be obtained (I3, the number of nodes on which statistics are collected--the number of sink nodes).

The fields are placed consequtive on a card with at most 25 values/card.

DATA CARD 4

Field 1 — The number of branches required to release each node (25I3, the largest node number assigned minus 1).

A value must be given for each number starting with node 2 and going to the largest node number assigned. Even if there is no node number 3, a value (or blank) must be given to the number of releases associated with node number 3. Node numbers must be greater than or equal to 2 and the first value on card number 4 is the number of releases associated with node number 2.

DATA CARD 5

Field 1 — The node type of each node. As in data card 4, all numbers from 2 to the largest node number assigned must be given node type designators (25I3, the largest node number assigned minus 1).

The characteristics associated with each node type are:

Node Type	When realized should scheduled events be removed	Reset number of releases	Output Type *
1	no	to	D
2	no	to	P
3	no	no	D
4	no	no	P
5	no	yes	D
6	no	yes	P
7	yes	yes	D
8	yes	yes	P

*D = Deterministic
P = Probabilistic

DATA CARD 6

Field 1 — The lower limit for the histogram to be obtained for each node on which statistics are being gathered. The lower limits are put on the card in the same order as the nodes are listed in Fields 2 and 3 of Data Card 3. The value assigned will be the lower limit of the second cell of the histogram. The first cell of the histogram will contain the number of times the node was realized in the time lower than the lower limit given (8F10.2, the number of nodes on which statistics are to be collected.)

DATA CARD 7

Field 1 — The width of each cell of the histogram. The widths are put in the same order as the nodes are listed in Fields 2 and 3 of Data Card 3 (8F10.2, the number of nodes on which statistics are to be collected.) Each histogram will contain 32 cells. The last cell will be fore values greater than the lower limit +30* (cell width).

DATA CARD 8

The parameters associated with the distribution of the time to perform each activity. One card is required for each activity with a different time characterization. The number of cards is specified by Data Card 1, Field 7. A maximum of 300 is permitted. The cards must be arranged by ascending parameter number and the parameters must be numbered consecutively or blank cards appropriately placed. Six distribution types are available in the GE 425 version of the GERT Simulation Program. These types are:

1. Constant
2. Normal
3. Uniform
4. Erlang
5. Lognormal
6. Poisson

The fields required are dependent on the distribution type of the activity. For distribution type 1 (Constant):

Field 1	The constant time (F10.4,1)

For distribution type 2 (Normal):

Field 1	The mean value (F10.4,1)
Field 2	The minimum value (F10.4,1)
Field 3	The maximum value (F10.4,1)
Field 4	The standard deviation (F10.4,1)

For distribution type 3 (Uniform):

Field 1	Not used (F10.4,1)
Field 2	The minimum value (F10.4,1)
Field 3	The maximum value (F10.4,1)
Field 4	Not used (F10.4,1)

For distribution type 4 (Erlang):

Field 1	The mean time for the Erlang variable divided by the value given to Field 4(F10.4,1)
Field 2	The minimum value (F10.4,1)
Field 3	The maximum value (F10.4,1)
Field 4	The number of exponential deviates to be included in the sample obtained from the Erlang distribution (F10.4,1)

If Field 4 is set equal to 1, an exponential deviate will be obtained from distribution type 4.

For distribution type 5 (lognormal):

Given:

μ_Q = mean of lognormal random variable

σ_Q^2 = variance of lognormal random variable

min Q = smallest sample allowable

max Q = largest sample allowable

Field 1	$\ln \mu_Q - \frac{1}{2}$ [value in Field 4]^2
Field 2	ln (min Q)
Field 3	ln (max Q)
Field 4	$\sqrt{\ln (\sigma_Q^2/\mu_Q^2 + 1)}$

The values in fields 1 and 4 are the mean and standard deviation of the normally distributed random variable associated with the lognormal random variable.

For distribution type 6 (Poisson):

Field 1	The mean minus the minimum value
Field 2	The minimum value
Field 3	The maximum value
Field 4	Not used

Care is required when using the Poisson since it is not usually used to represent an interval of time. The interpretation of the mean should be the mean number of time units per time period.

Samples are obtained from the distributions such that if a sample is less than the minimum value, the sample value is given the minimum value. Similarly, if the sample is greater than the maximum value, the sample value is assigned the maximum value. This is not sampling from a truncated distribution but sampling from a distribution with a given probability of obtaining the minimum and maximum values. The GERT Simulation Program has been written to facilitate the adding of other distribution types. FORTRAN subroutines for obtaining deviates from other distribution types are available (14,23).

DATA CARD 9

One data card is for each activity associated with the network.

Field 1	probability of realization (F8.3)
Field 2	Start node (I3,1)
Field 3	End node (I3,1)
Field 4	Parameter number (I3,1)
Field 5	The distribution type (I3,1)
Field 6	Count type (I3,1)
Field 7	Activity number (I3,1)

The last data card type must have a zero (or blank) in Field 2.

DATA CARD 10

Only required if number of nodes modified is greater than zero. (Field 7, Data Card 2)

Field 1	An activity number (I3,1)
Field 2	The number of the node to be replaced if the activity given in Field 1 is realized (I3,1)
Field 3	The number of the node to be inserted into the network in place of the node specified in Field 2 when the activity in Field 1 is realized (I3,1)
Fields 4-21	Fields 2 and 3 are repeated if the activity given in Field 1 affects multiple nodes. A zero in an even-numbered field indicates the end of the data on the card.

The last card of this type must have a zero in Field 1.

DATA CARD 11

Only used if the project number is negative. (Field 2, Data Card 1)

Field 1	The run number for which tracing of the end of activity events should begin (I3,1)

Field 2	The run number for which tracing of the end of activity events should terminate (I3,1)

Multiple networks can be analyzed by stacking the data cards as described above, one after another. No blank cards should separate the data cards for each network. A blank card is required to indicate the end of all networks to be simulated.