

FACILITY FORM 602

N 50	(ACCESSION NUMBER)		(THRU)
38	(PAGES)		NINE
CR 96 321	(NASA CR OR TMX OR AD NUMBER)		(CATEGORY)

Reproduced by
**NATIONAL TECHNICAL
INFORMATION SERVICE**
US Department of Commerce
Springfield, VA. 22151

USER MANUAL
for
GERT EXCLUSIVE-OR PROGRAM

Research Sponsored
by
Electronics Research Center
National Aeronautics and Space Administration
NASA Research Grant NGR 03-001-034

Phillip C. Ishmael and A. Alan B. Pritsker



38

ABSTRACT

This report describes the procedures for using digital computer programs for analyzing GERT networks which contain nodes of the EXCLUSIVE-OR type and branches which have both a probability and a time associated with them. The time associated with a branch can be a random variable. The program calculates the probability, the expected time and the variance in the time to go from each source node of the GERT network to each sink node.

Programs have been written in FORTRAN II and FORTRAN IV and have been exercised on the IBM 1130, GE 225 and CDC 3400 computers. This report details the operating procedures for using these programs. Four examples are included to illustrate the important facets involved in using the programs. Special options included in the program are:

- (1) printout of all loops and paths associated with the network; and
- (2) deletion of loops that have a probability of occurrence less than a value specified by the user.

The program described in this report has been submitted to COSMIC.

USER MANUAL FOR
GERT EXCLUSIVE-OR PROGRAM

The procedure for using a digital computer program for analyzing GERT networks containing only EXCLUSIVE-OR nodes is presented in this report. The programs are modifications of the original program developed at The RAND Corporation by A. B. Nelson. Background information on GERT is not included in this report. A bibliography on GERT and related topics is given at the end of this report.

Three FORTRAN versions of a digital computer program are available for obtaining pertinent information for GERT networks containing only EXCLUSIVE-OR nodes. The utilization procedure described in this report applies to the GE 225 FORTRAN II version and the IBM 1130 FORTRAN IV version. The input to these programs is identical (even though the two FORTRAN versions are slightly different). The FORTRAN IV version has been run on the CDC 3400.

The program determines the source nodes, the sink nodes, the paths connecting the source nodes to the sink nodes, and the loops of the network. The standard output from the program includes: (1) appropriate problem identification headings; (2) the paths and loops of a network; (3) the probability of realizing a sink node from any source node; and (4) the mean and variance of the time to realize a sink node, given that the sink node is realized and given an initial source node. The option exists to delete the loop and/or path output for large or complex networks if it is so desired. The exercising of this option

can be beneficial on machines having relatively slow printers (such as the IBM 1130) or on high speed printers where the standard operating practice is to have a relatively low print limit.

Input to the program includes appropriate problem identification information and the branches of the network. Information concerning each branch includes the start node and end node for the branch, the probability of realizing the branch, and data about the moment generating function (M.G.F.) of the random variable associated with the branch. The M.G.F. is described by a three-letter code and up to two parameters of the M.G.F. The program determines all paths and loops of the network based on the input information. The desired output statistics are computed using the values associated with the loops and paths of the network.

The reader is referred to the GERT report (15, pp. 81-84) for information concerning the method for calculating the output statistics. The definition of variables and the techniques employed in the program are described in a separate report (11). This report describes the input procedure, the output formats and the options available to the user. Examples to illustrate these facets of the program are presented.

Program Operating Procedure

The input specification for the first data card to the program and the first data card for each network are given in Table 1. Each branch of a network must be described by a separate card as shown in Table 2. The general arrangement of the input data cards is shown in

Table 1 - Input to GERT Program

FIRST CARD OF THE DATA INPUT DECK

The first card of the input deck is the means by which the distribution codes are placed in the machine. These codes are then used to check the distribution codes in the input network.

Field 1 (cc. 1-9) = ABDEGNOU

FIRST CARD OF EACH INPUT NETWORK IN THE INPUT DECK (HEADER CARD)

The first card of each input network contains the user name and the problem identification. Also contained in this card are the three program control options available to the user.

Field	Card Columns	Definition
1	1-4	Blank
2	5-14	User name
3	17-20	Problem designation (right justified) may be any combination of alphabetic or numeric characters
4	21-22	Month (right justified)
5	23-24	Day (right justified)
6	25-28	Year
7	29-30	Loop printout control option: If blank, print loops; If positive, do not print print loops
8	31-32	Path printout control option: If blank, print paths; If positive, do not print paths
9	33-42	Loop deletion probability (F10.8); if this is left blank, no loops will be deleted
10	43-44	Option for adjustment of the equivalent branch printout when the loop deletion option is exercised; if this is left blank, the actual calculated branch values are printed. If this field contains a positive number, the equivalent branch printout will be adjusted so that the probabilities for all equivalent branches emanating from a given source node sum to one.

Table 2 - Input to GERT Program for Each Branch of the Network

Each network is specified by defining its branches as shown below:

1. Node beginning the branch;
2. Node terminating the branch;
3. Type of distribution of time associated with the branch;
4. Probability of realizing the branch if its beginning node is realized; and
5. Coefficients defining the time distribution.

One data card is required for each branch of the network. The order of the data cards is not significant. The fields on each data card are:

Field	Card Columns	Definition
1	1-4	Node beginning branch (right justified)
2	6-9	Node terminating branch (right justified)
3	11-13	Type of distribution [B, D, E, GA, GE, NB, NØ, P, U] (left justified)
4	14-20	The definition of these fields depends on the type of distribution (see below). The format for all fields is F7.3.
5	21-27	
6	28-34	
7	35-41	
8	42-48	
9	49-55	
10	56-62	
11	63-69	

Type of Distribution *	Field								
	4	5	6	7	8	9	10	11	
B (Binomial)	Prob.	n	p	-	-	-	-	-	
D (Discrete)	Prob. 1	T1	Prob. 2	T2	Prob. 3	T3	Prob. 4	T4	
E (Exponential)	Prob.	1/a	-	-	-	-	-	-	
GA (Gamma)	Prob.	1/a	b	-	-	-	-	-	
GE (GEometric)	Prob.	p	-	-	-	-	-	-	
NB (Neg. Binomial)	Prob.	r	p	-	-	-	-	-	
NO (NORmal)	Prob.	m	σ	-	-	-	-	-	
P (Poisson)	Prob.	λ	-	-	-	-	-	-	
U (Uniform)	Prob.	a	b	-	-	-	-	-	

*See Table 3 for Definition of the Parameters

Fig. 1. The equations and moments of the distributions that have been programmed are shown in Table 3. Since only the mean and variance of each branch is used in the calculation of the mean and variance of the system, other distributions can be accommodated by the program by specifying the normal distribution with mean and standard deviation values of the distribution of interest (1, p. 87). Further, it can be shown that n^{th} central moments for the network only depends on j central moments of the branches, $j=1, 2, \dots, n$. This should aid in developing programs for computing higher moments. The GE 225 and IBM 1130 versions of the program will accept an input network containing up to one hundred branches although, as will be demonstrated by the first example problem, storage limitations can result in a smaller problem being too large for the computer if there are many possible paths through a network. When this occurs, an error message is printed out and execution of the problem is terminated. The only restrictions placed on the numbering of nodes is that all node numbers must be positive integers greater than zero and ≤ 100 . Node numbers can be in any order, i.e., they do not have to be in ascending order as in the case in most CPM programs.

For the computer program only one branch can join a given start and end node. There can, however, be another branch between the two nodes if the roles of the start node and end node are reversed as is shown below:

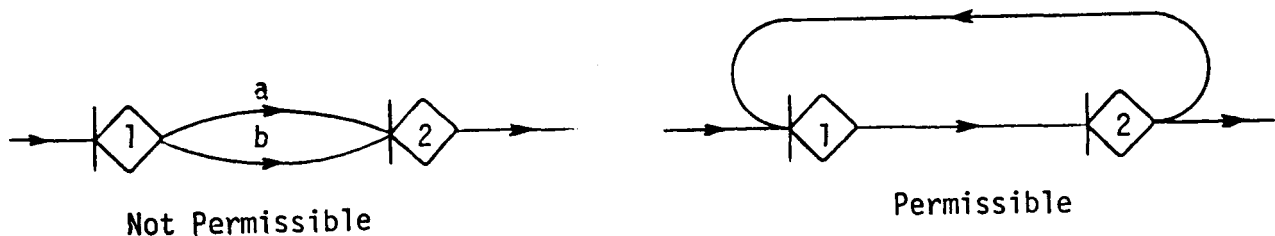


Table 3 - Distribution Acceptable to GERT Program

Type of Distribution	$M_E(s)$	Mean	Second Moment	Input Variables
Binomial (B)	$(pe^s + 1 - p)^n$	np	$np(np + 1 - p)$	$w_E(o); n, p$
Discrete (D)	$\frac{p_1 e^{st_1} + p_2 e^{st_2} + \dots}{p_1 + p_2 + \dots}$	$\frac{p_1 T_1 + p_2 T_2 + \dots}{p_1 + p_2 + \dots}$	$\frac{p_1 T_1^2 + p_2 T_2^2 + \dots}{p_1 + p_2 + \dots}$	$w_E(o); p_1, T_1, p_2, T_2;$
Exponential (E)	$(1 - s/a)^{-1}$	$1/a$	$2/a^2$	$w_E(o); 1/a$
Gamma (GA)	$(1 - s/a)^{-b}$	b/a	$\frac{b(b+1)}{a^2}$	$w_E(o); 1/a, b$
Geometric (GE)	$\frac{pe^s}{1 - e^s + pe^s}$	$1/p$	$\frac{2-p}{p^2}$	$w_E(o); p$
Negative Binomial (NB)	$\left(\frac{p}{1 - e^s + pe^s} \right)^r$	$\frac{r(1-p)}{p}$	$\frac{r(1-p)(1+r-rp)}{p^2}$	$w_E(o); r, p$

(Continued)

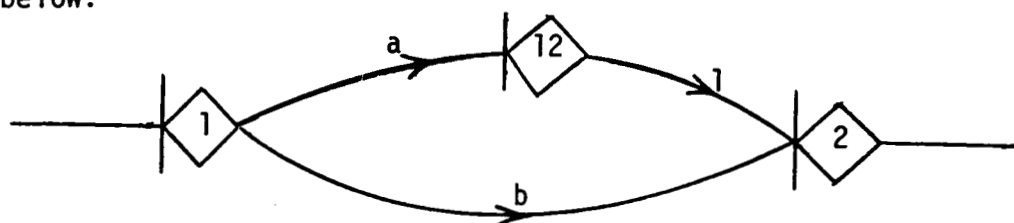
Table 3 - Distribution Acceptable to GERT Program
(Continued)

Type of Distribution	$M_E(s)$	Mean	Second Moment	Input Variables
Normal (NØ)	$e^{(sm + \frac{1}{2}s^2\sigma^2)}$	m	$m^2 + \sigma^2$	$w_E(o); m, \sigma$
Poisson (P)	$e^{\lambda(e^s - 1)}$	λ	$\lambda(1 + \lambda)$	$w_E(o); \lambda$
Uniform (U)	$\frac{e^{sa} - e^{sb}}{(a-b)s}$	$\frac{a+b}{2}$	$\frac{a^2 + ab + b^2}{3}$	$w_E(o); a, b$

If the network of interest has more than one branch between the same two nodes as shown on the previous page on the left, then either:

(1) the techniques discussed in the GERT report (15, pp. 20, 32) should be used to reduce the parallel paths to their equivalent single path prior to inputting the network into the computer program; or

(2) a dummy node should be inserted for one of the branches as shown below:



Program control options are included to allow the user to decide whether or not the loops and paths of the network are to be printed. For networks having a large number of loops and paths it may be beneficial to delete the printing of the loops and/or paths. The nodes associated with loops and paths are printed depending on the values inserted into Fields 7 and 8 of the heading card associated with a network. An option to delete higher order loops which have a low probability of occurrence can be exercised through Field 9 of data card 1. If this option field is left blank, all loops will be considered. If, however, loops should be deleted that have a probability of realization less than or equal to δ then δ can be read in as the deletion probability in Field 9 and no loops with lower probabilities will be included. The effect upon the accuracy of the results for the equivalent network will depend upon the size the complexity of the network as well as the magnitude of δ . An example of the effect of

various δ levels is included as a part of the first example problem. Since the deletion of higher order loops having low probability of occurrence introduces some error into the final output, the probabilities for all equivalent branches of the network emanating from a given source node may not sum to one. The user can cause the program to adjust the probabilities so that they sum to one by the value specified in Field 10 of the first data card. The mean and variance of the time to traverse the equivalent branch are adjusted by the dividing by the nonadjusted probabilities.

Multiple networks can be analyzed by separating the data decks with a blank card. The last data deck should be followed by a blank card and a card with a negative value in columns 1-4. Figure 1 portrays the organization of the data for analyzing two networks.

Examples

Four example problems will be presented. The first problem illustrates in detail the output of the GERT program. This example starts with a simple network which is expanded in a modular manner to illustrate the use and effect of some of the program options. Run times will also be given for some of the problem variations to indicate what might be expected for similar problems. No definite rule has been established for run time required for networks with a given number of input branches since the run time depends strongly on the nature of the network in addition to the number of branches of the network. The second example is a three-player game in which the game is won by the first player to win two consecutive games. The third example is a model of a manufacturing process and the fourth example is for a two-port related to a portion of a launch vehicle under test during countdown. Only the final results are presented for the last three example problems.

Example 1

The network for this example (15, pp. 87-90) is shown in Fig. 2. The input data for the network is shown in Table 4. The printouts from the program for this example will now be described. The first output is an echo check of the input network as shown in Table 4. The "D" code in Field 3 indicates that all times for this example are discrete. (See Table 2). It should be noted that, given the echo check, the input network can be reconstructed. Also, since Fields 7, 8, 9, and 10 of the heading card are blank, all loops and paths will be printed and no higher order loops will be deleted.

Intermediate output available from the program is shown in Tables 5 through 7. Table 5 presents the input data for each branch in terms of its mean and variance. Table 6 gives a listing of the loops of the network. The order of the loop and the probability associated with the loop are given on one line. The component loop information is recorded below this line. The code $w(0)$ is the probability associated with the loop or component loop. After the word NODES, the nodes in the component loops are printed, e.g., 3 5 indicates a loop with branches going from node 3 to node 5 and then node 5 to node 3. The output variables NS and NE in Table 7 are the start and end nodes for each path. The probability of realizing a given path is listed under "PROB" and the mean and variance of the time to traverse a path given that it is realized are listed as "M T" and "V T," respectively. On the right on the next line, the nodes of the path are then listed. The final results are shown in Table 8. In Table 8, the source nodes of the network are listed under "ENTRY" and the sink nodes under "EXIT."

	5	6	7	10	15	20	25	30	35	40	45	50	55	60	65	70	72	73	80
1	A	B	E	G	N	O	P	U											
2	PHIL	ISHMAEL	EL			6/14/1968													
3	3	D			1	0													
4	4	D			1	0													
5	5	D			0	6													
6	6	D			0	4													
7	5	D			0	3													
8	6	D			0	7													
9	3	D			0	2													
10	4	D			0	3													
11	7	D			0	8													
12	8	D			0	7													
13	BLANK	CARD																	

GERT PROBLEM NO. 1 BY PHIL ISHMAEL DATE 6/14/1968

INPUT NETWORK		1	3	D	1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	4	D	1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	5	D	0.600	2.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	6	D	0.400	4.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
4	5	D	0.300	5.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
4	6	D	0.700	2.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
5	3	D	0.200	7.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
6	4	D	0.300	6.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
5	7	D	0.800	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
6	8	D	0.700	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 4 - Input Data Cards for Example 1 and Echo Check of the Input Network

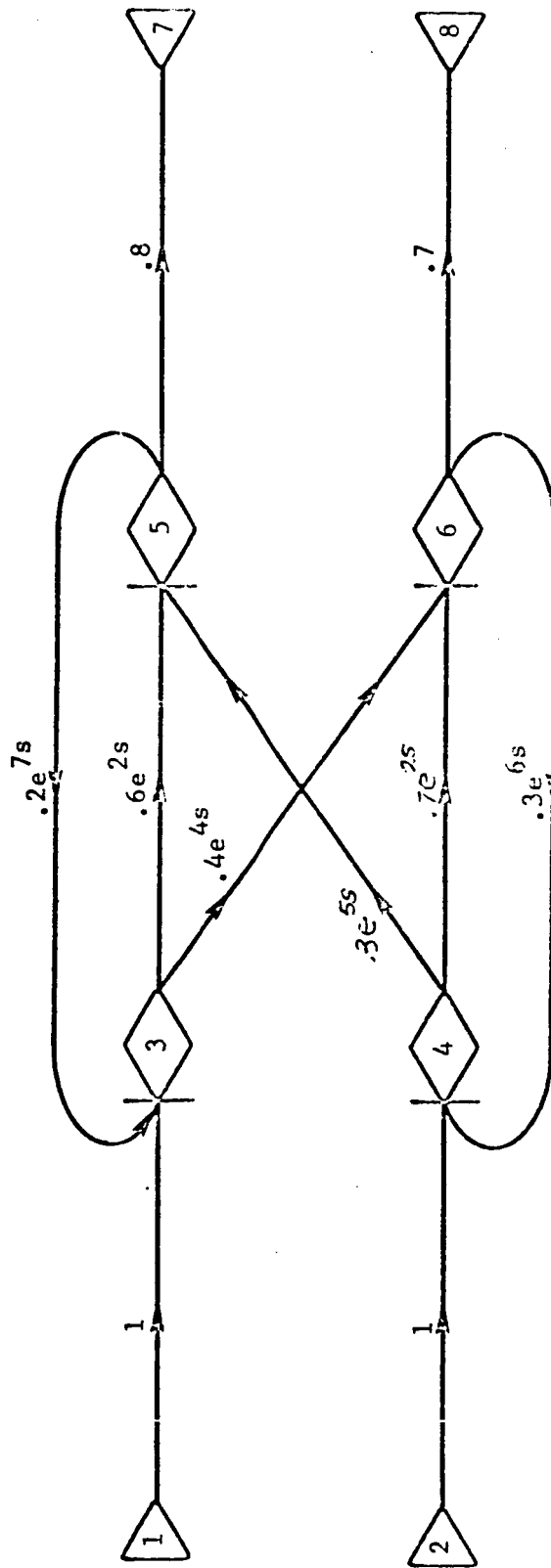



Fig. 2 - GERT Network for Example 1

Table 5 - Calculation of Branch Parameters

INPUT NETWORK				
NODES AND PROBABILITY OF SELECTION WITH MEAN AND VARIANCE OF TIME FOR EACH LINK				
FROM	TO	PROB	MEAN	VAR
1	3	1.000	0.000	0.000
2	4	1.000	0.000	0.000
3	5	0.600	2.000	0.000
3	6	0.400	4.000	0.000
4	5	0.300	5.000	-0.000
4	6	0.700	2.000	0.000
5	3	0.200	7.000	-0.000
6	4	0.300	6.000	-0.000
5	7	0.800	0.000	0.000
6	8	0.700	0.000	0.000

Table 6 - Listing of Network Loops *

LOOP OF ORDER	1	W(O)= 0.119999			
		W(O)= 0.1199 , NODES	3	5	
LOOP OF ORDER	2	W(O)= 0.025199			
		W(O)= 0.1199 , NODES	3	5	
		W(O)= 0.2099 , NODES	4	6	
LOOP OF ORDER	1	W(O)= 0.007199			
		W(O)= 0.0071 , NODES	3	6	4 5
LOOP OF ORDER	1	W(O)= 0.209999			
		W(O)= 0.2099 , NODES	4	6	

Reproduced from
best available copy. 

*This table is taken from the IBM 1130 Computer which prints only the columns desired and does not round up.

Table 7 - Listing of Network Paths

NS	NE	PROB	M T	V T				
1	7	0.551162	3.4926	19.7059	1	3	5	7
1	7	0.041860	18.6191	41.2408	1	3	6	4
1	8	0.406976	7.6191	41.2409	1	3	6	8
2	8	0.024418	19.6191	41.2409	2	4	5	3
2	7	0.348837	8.6191	41.2409	2	4	5	7
2	8	0.626744	4.3919	28.6892	2	4	6	8

Table 8 - Final Results for Example 1

GERT PROBLEM NO. 1 BY PHIL ISHMAEL DATE 6/14/1968

LOOP DELETION VALUE, DEL = 0.00000000

EQUIVALENT BRANCHES OF THE NETWORK

ENTRY	EXIT	PROBABILITY	MEAN TIME	VARIANCE
1	7	0.593023E 00	0.456036E 01	0.362374E 02
1	8	0.406976E 00	0.761918E 01	0.412409E 02
2	8	0.651162E 00	0.496293E 01	0.375289E 02
2	7	0.348837E 00	0.861918E 01	0.412409E 02

Execution time for this program on both the GE 225 and the IBM 1130 was in the neighborhood of one minute. It should be mentioned that the times given for this and the remaining examples are only approximate since neither machine has an internal clock. The times were recorded by machine operators and are probably somewhat larger than was required.

In order to further check run time, additional problems were run using series of "modules" based on the network of Fig. 2. Run time was not appreciably increased by the two-module test problem. However, the three-module problem shown in Fig. 3 did increase the running time. Five cases were considered and are labeled 1A, 1B, 1C, 1D and 1E. The first four cases 1A through 1D involved different values for the deletion probabilities as shown by Field 9 for the first four problem heading cards in Table 9. Problem 1E illustrates the use of the option to adjust the equivalent probabilities (a 1 in Field 10), and the option to not print loops and paths (ones in Fields 7 and 8). Table 10 shows the effect of the various deletion values upon the final results for the network. Problem 1A is, of course, the most accurate one since no higher order loops were deleted. As expected, accuracy decreases with an increase in the deletion probability. The normalizing of the values (case 1E) compensates for the deletion of loops but only partially. Table 11 shows the effect of the deletion probability on the number of higher order loops.

Run time for the network of Fig. 3 is approximately 4 minutes for the GE 225. If all loops and paths are printed, the run time on the IBM 1130 is 8 minutes. By exercising the option to omit the printout of

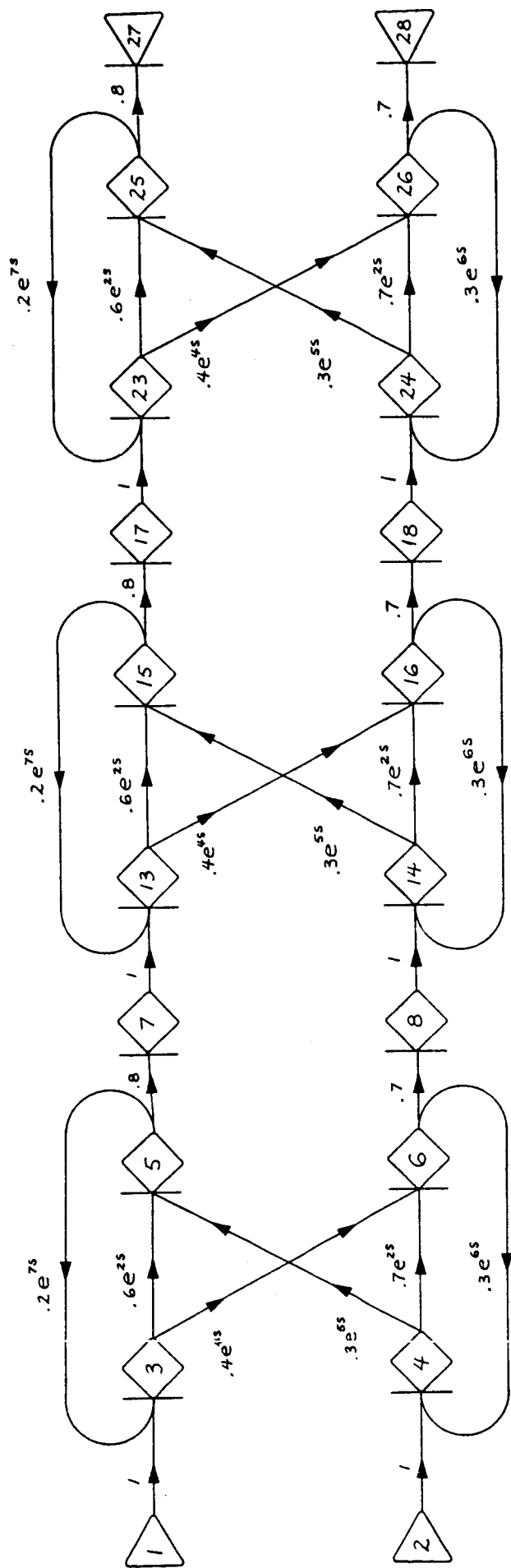


Fig. 3 - Three-Module Problem.

Table 9 - Header Cards for the Five Problems

Phil Ishmael	1A	4291968	0 0	.0
Phil Ishmael	1B	4291968	0 0	.00001
Phil Ishmael	1C	4291968	0 0	.00005
Phil Ishmael	1D	4291968	0 0	.0001
Phil Ishmael	1E	4291968	1 1	.0001 1

Table 10 - Equivalent Branches of the Network

ENTRY	EXIT	PROBLEM	PROBABILITY	MEAN TIME	VARIANCE TIME
1	27	1A	0.469389E 00	0.177158E 02	0.127568E 03
		1B	0.469346E 00	0.177100E 02	0.127190E 03
		1C	0.469839E 00	0.177710E 02	0.130857E 03
		1D	0.470503E 00	0.178136E 02	0.132018E 03
		1E	0.469787E 00	0.177867E 02	0.131815E 03
1	28	1A	0.530634E 00	0.178962E 02	0.120472E 03
		1B	0.530586E 00	0.178905E 02	0.120094E 03
		1C	0.531142E 00	0.179515E 02	0.123761E 03
		1D	0.531022E 00	0.179398E 02	0.123131E 03
		1E	0.530213E 00	0.179125E 02	0.122942E 03
2	28	1A	0.545194E 00	0.179425E 02	0.125641E 03
		1B	0.545145E 00	0.179367E 02	0.125263E 03
		1C	0.545870E 00	0.180078E 02	0.129291E 03
		1D	0.545211E 00	0.179658E 02	0.127725E 03
		1E	0.545008E 00	0.179592E 02	0.127675E 03
2	27	1A	0.454829E 00	0.188962E 02	0.120472E 03
		1B	0.454788E 00	0.188905E 02	0.120094E 03
		1C	0.455265E 00	0.189515E 02	0.123761E 03
		1D	0.455162E 00	0.189398E 02	0.123131E 03
		1E	0.454992E 00	0.189329E 02	0.123084E 03

Table 11 - Effects of the Deletion Probability Upon the Number of Higher Order Loops for the Network of Fig. 3.

Loop of Order	DEL = 0.000000 No. of Loops	DEL = 0.00001 No. of Loops	DEL = 0.00005 No. of Loops	DEL = 0.0001 No. of Loops
1	9	9	9	9
2	30	30	30	27
3	45	41	38	38
4	30	27	15	15
5	9	6	6	3
6	1	1	0	0

loops and paths, the run time on the IBM 1130 is reduced significantly. A four-module problem that was run on the CDC 3400 required only 36 seconds for execution.

The network of Fig.3 is more complex than it appears. Actually it is a good check for program capability since its analysis involves a total of 124 loops and 54 paths. The four-module problem mentioned above is too large for the standard GE 225 and IBM 1130 versions of the GERT program even though there are only 40 branches in the input network. The reason for this is that the dimensioned variable LOOP(I) that keeps track of all nodes in first-order loops and all nodes comprising paths through the network is only dimensioned at 1000 whereas it becomes much larger than that for a four-module network due to the large number of paths each containing several branches of the network. Although the GE 225 and IBM 1130 versions should handle many problems having more than 40 branches in the input, when a program does become too large an error message will be printed and execution will be terminated. If no larger computer is available, reference to the companion manual (11) which contains a detailed description of how the program works will indicate the modifications to the program which could allow the problem of interest to be run. Table 12 shows how the number of loops and paths increase as the number of modules are increased.

Table 12 - Number of Loops and Paths for a Series of 1 Through 4 Modules of the Network of Fig. 2.

Number of Modules	Number of Loops	Number of Paths	Number of Values of LOOP(I)
1	4	6	45
2	24	18	190
3	124	54	735
4	624	162	2746

Example 2

The GERT program was used to analyze the three-player game (15, pp. 91-93) depicted by the network shown in Fig. 4. Three players (denoted by A, B, and C) take turns playing a game according to the following rules: at the start A and B play while C sits out; the winner of the match between A and B then plays C; the winner of the second match then plays the loser of the preceding match until a player wins twice in succession, in which case he is declared the winner of the game. It is assumed in this analysis that $p_{AB} = p_{BC} = p_{CA}$; that is, the probability of A beating B, of B beating C, and C beating A are equal. Also, the time element is assumed to be one for each game played. The GERT program was run to analyze the game when the independent variable was p_{AB} . The input data for this example for one run ($p_{AB} = 0.20$) is shown in Fig 5. Note on the header card that the intermediate printout of loops and paths was not requested. The computer output for the case $p_{AB} = 0.20$ is given in Table 13. Summary results of the program are shown in Table 14 for various values of p_{AB} . Only values of $p_{AB} \leq .5$ are tabulated because the game is symmetrical.

The results shown in Table 14 indicate that player A has the highest probability of winning the game for values of p_{AB} in the range from 0.10, to 0.50. Also, as p_{AB} moves from the value 0.50 toward the value 0.10, the expected number and variance of games played before a winner is declared increases. The run time for the network of Fig. 4 averaged about 0.75 minutes per run for the seven runs required.

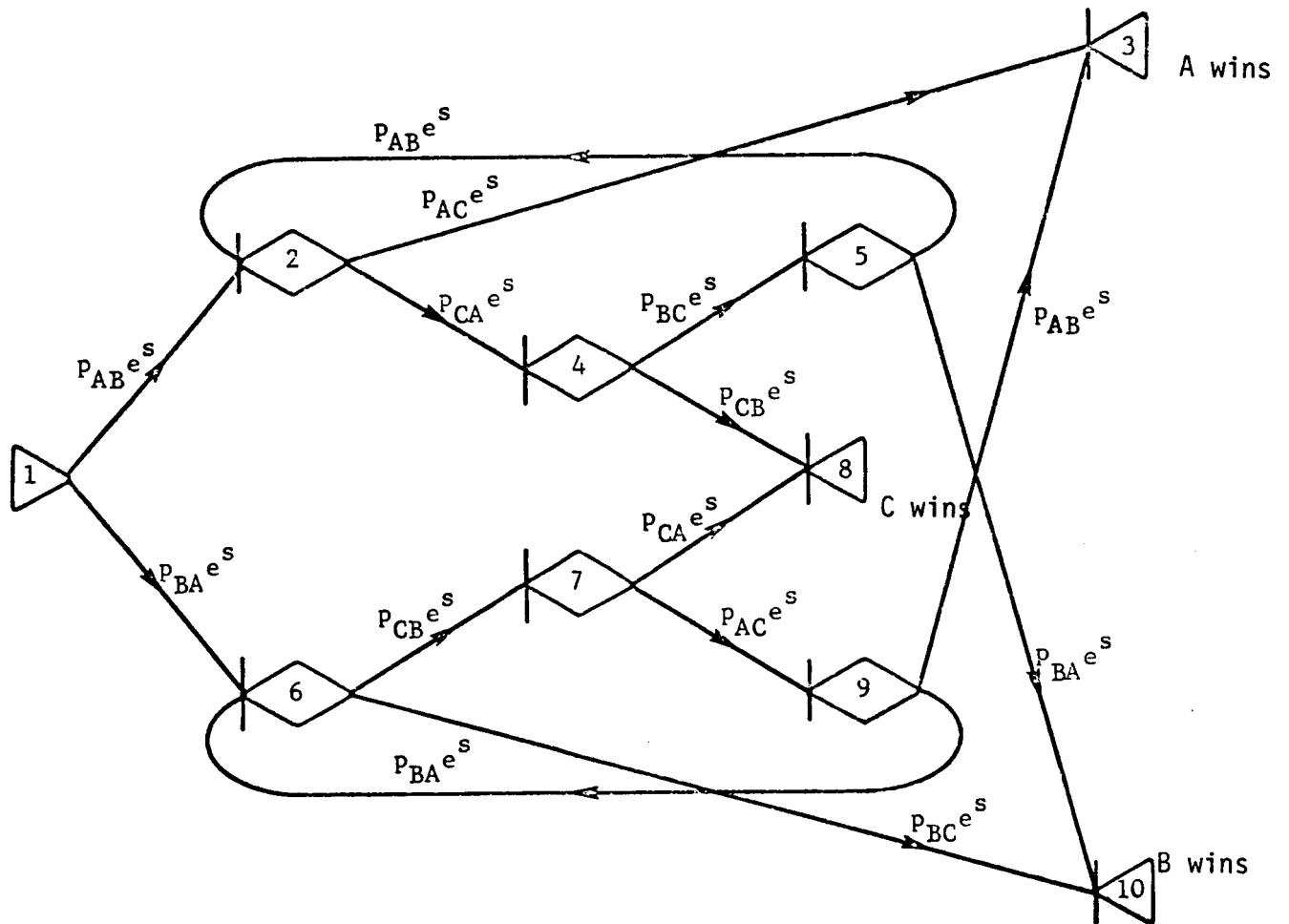


Fig. 4 - GERT Network of a Three-Player Game

FORTRAN CODING FORM

NAME <u>Phil Ishmael</u>	COURSE, SECTION
PROBLEM <u>GERT Problem No. 22</u>	DATE <u>5-16-68</u>
COMPILER <u>IBM 1130</u>	INSTRUCTOR

STATEMENT NUMBER	FORTRAN STATEMENT	55	60	65	70	75	80
1	ABDEGNOPU						
2	PHIL ISHMAEL 22 5161968						
3	2. D 0.2						
4	6. P 0.8						
5	3. P 0.8						
6	4. P 0.2						
7	5. P 0.2						
8	8. D 0.8						
9	10. P 0.8						
10	2. D 0.2						
11	7. D 0.8						
12	10. P 0.2						
13	8. P 0.2						
14	9. P 0.8						
15	3. P 0.2						
16	6. D 0.8						
17	BLANK CARD						
18							
19							

Fig. 5 Input Data for Example 2

Table 13 - Computer Output for Example 2 with $P_{AB} = 0.20$

GERT PROBLEM NO. 22 BY PHIL ISHMAEL DATE 5/16/1968					
LOOP DELETION VALUE, DEL = 0.0000000					
EQUIVALENT BRANCHES OF THE NETWORK					
ENTRY	EXIT	PROBABILITY	MEAN TIME	VARIANCE	
1	3	0.371126E 00	0.492094E 01	0.17422CE 02	
1	10	0.334320E 00	0.512586E 01	0.190015E 02	
1	8	0.294553E 00	0.580548E 01	0.181898E 02	

Reproduced from
best available copy.



Example 3

The third example is a model of a manufacturing process (15, pp. 57-60). On a production line a part is manufactured at the beginning of the time. The manufacturing operation is assumed to take four hours. Before the finishing operations are started on the part, it is inspected--with 25 percent of the parts failing the inspection and requiring rework. The inspection time (including time spend waiting for inspection) is assumed to be distributed according to the negative exponential distribution with a mean of one hour. Reworking requires three hours, and 30 percent of the reworked parts fail the next inspection. The inspection of reworked items is also distributed according to the negative exponential with a mean of one-half hour. Parts which fail this second inspection are scrapped. If the part passes either of the first two inspections, it is sent to the final finishing operation which takes 10 hours 60 percent of the time and 14 hours 40 percent of the time. A final inspection, which

Table 14 - Results of Three-Player Game with $P_{AB} = P_{BC} = P_{CA}$

P_{AB}	A Wins			B Wins			C Wins		
	Prob.	Expected No. of games	Variance	Prob.	Expected No. of games	Variance	Prob.	Expected No. of games	Variance
0.10	0.3591	9.5445	85.9734	0.3330	10.0537	89.1947	0.3079	10.8341	88.5718
0.20	0.3711	4.9209	17.4220	0.3343	5.1259	19.0015	0.2946	5.8055	18.1898
0.30	0.3724	3.5479	6.1122	0.3391	3.5958	6.7711	0.2885	4.2334	5.9870
0.35	0.3702	3.2087	4.0221	0.3427	3.2206	4.4116	0.2871	3.8456	3.6645
0.40	0.3665	2.9931	2.8553	0.3471	2.9895	3.0647	0.2862	3.6039	2.3538
0.45	0.3621	2.8707	2.2658	0.3520	2.8653	2.3554	0.2858	3.4710	1.6783
0.50	0.3571	2.8286	2.1094	0.3571	2.8286	2.1094	0.2857	3.4286	1.4694

takes one hour, rejects 5 percent of the parts, which are then scrapped. The GERT network for this process is shown in Fig. 6

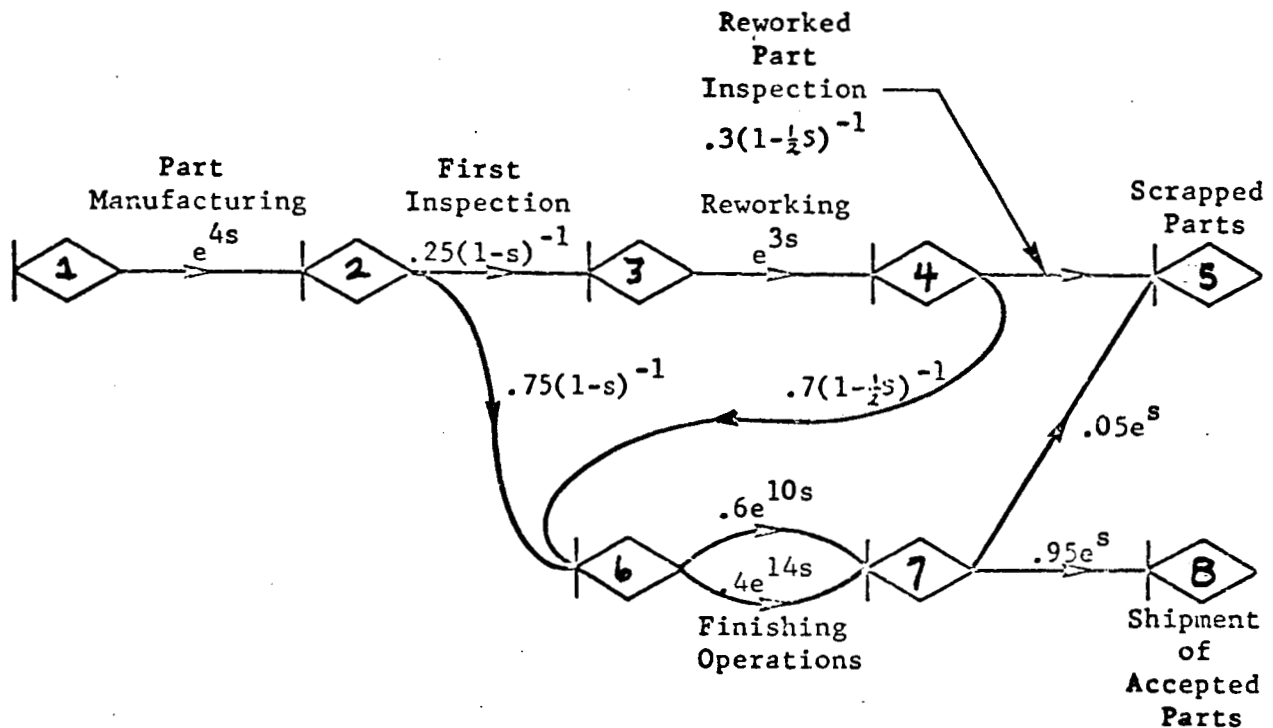


Fig. 6 - GERT Network of a Manufacturing Operation

The network of Fig. 6 has parallel paths between nodes 6 and 7. This must be modified as described earlier. The equivalent M.G.F. for the above two parallel branches is: $M_E(s) = 0.6e^{10s} + 0.4e^{14s}$. Thus two probabilities and two times must be inputted on the same input card for nodes 6 and 7 as described in Table 1 for discrete distributions. An alternate approach to resolving the parallel branches is to add a dummy node as shown in Fig. 7.

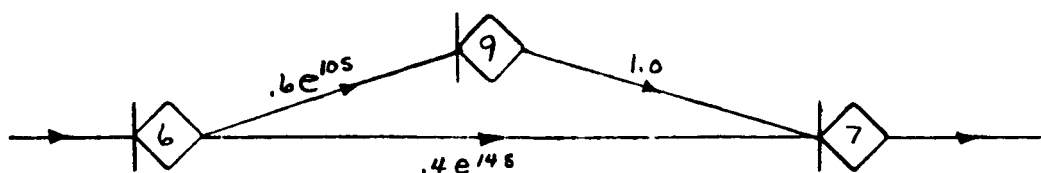


Fig. 7 - Adding a Branch to Eliminate Parallel Branches Between Nodes

Both methods were applied to this problem and identical results obtained.

The input data for Example 3 with the parallel branches reduced to an equivalent branch is given in Fig. 8. The computer results for this network are given in Table 15. The proportion of parts being scrapped (Node 5) is about 12 percent and the proportion of good parts (Node 8) is about 88 percent. The mean processing time required for good parts is about 18 hours with a variance of about 7 hours. The variance for scrapped parts is quite large in comparison to the mean and is due to the large relative difference in the time required to scrap a part. On the average, some parts are scrapped after eight and one half hours while others receive 17 or 18 hours of processing before they are scrapped.

Table 15 - Equivalent Branches of the Network of Example 3

GERT PROBLEM NO. 31 BY PHIL ISHMAEL DATE 5/ 17/ 1968				
LOOP DELETION VALUE, DEL = 0.00000000				
EQUIVALENT BRANCHES OF THE NETWORK				
ENTRY	EXIT	PROBABILITY	MEAN TIME	VARIANCE
1	5	0.121249E 00	0.122237E 02	0.258396E 02
1	3	0.878749E 00	0.192621E 02	0.676663E 01

FORTRAN CODING FORM

NAME	Phil Ishmael		
PROBLEM	GERT Problem No. 31	COURSE, SECTION	
COMPILER	IBM 1130	DATE	5-17-68
		INSTRUCTOR	

STATEMENT NUMBER	FORTRAN STATEMENT	70	72	73	75	77	79	80	Identification
1	ABDEGNOPU								
2	PHIL ISHMAEL	31	5171968						
3	2 D	1.00	4.0						
4	3 E	0.25	1.0						
5	6 E	0.75	1.0						
6	4 D	1.00	3.0						
7	5 E	0.30	0.5						
8	6 E	0.70	0.5						
9	7 P	0.60	10.0	0.4	14.0				
10	5 D	0.05	1.0						
11	8 D	0.95	1.0						
12	BLANK CARD								
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									
25									
26									
27									
28									
29									
30									

Fig. 8 Input Data for Example 3

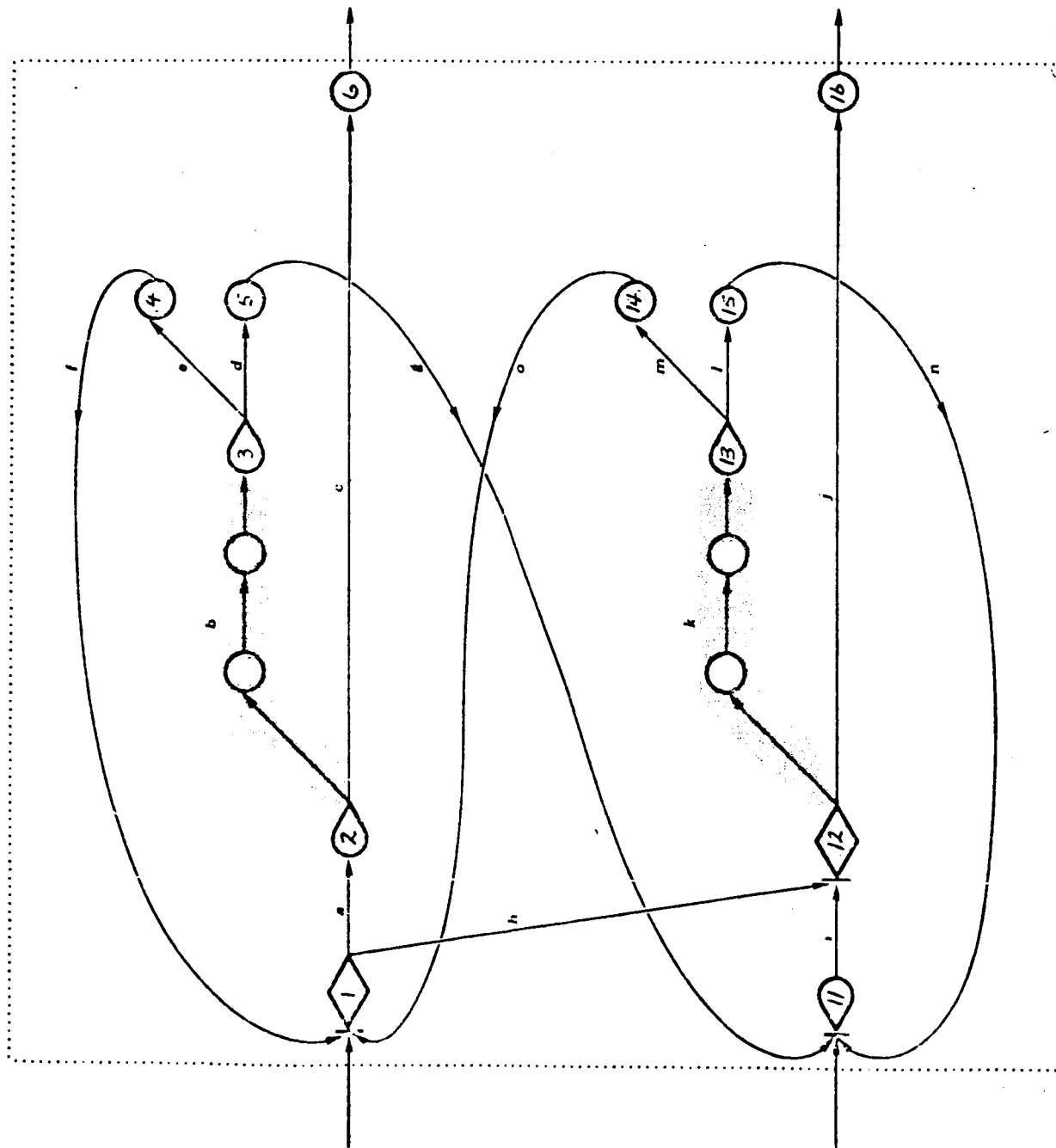
Example 4

The fourth example is taken from a study of a space vehicle count-down (3, pp. 39-42). The example deals with a two-port which is defined as follows (3, p. 22):

"A two-port is a network representation of a portion of the vehicle under test. It defines the possible transformation an activity can cause during countdown. Such an activity can be a means of changing module status. The changes are both planned and unplanned. For countdown analysis, the status variables of interest are the time before and after the test, and the module conditions. The time variable describes activity duration. Given the starting time, the activity's termination time can be derived. The module's condition can be considered as an activity input that is possibly changed by activity performance. The module's condition will be adequately described by two possible states--no undiscovered malfunctions, and undiscovered malfunctions. Since there are two input and two output states, the term two-port is used to describe the count-down activity."

The network for the present problem is shown in Fig. 9. The activities of interest are designated by lower-case alphabetical characters. Some of the nodes are not numbered because they represent an additional breakdown of activities that are aggregated at a higher level for this example. Thus activities b and k each represent three activities but they are considered to be only one for this problem. Even though some of the nodes of Fig. 6 are not EXCLUSIVE-OR nodes, the GERT program can be used because no more than one branch ever goes into the AND side of a node.

Table 16 contains numerical values for six arbitrary two-ports. Two dummy start nodes are used when inputting the program since nodes 1 and 11 both have branches looping back to them; however, in discussing the output, the equivalent branches will be described as if nodes 1 and 11 were the actual start nodes.



- a = No failures generated
- b = Fault malfunction indicated, diagnose, repair
- c = No malfunction indicated
- d = Undiscovered failures
- e = No undiscovered failures
- f = Recovery Policy 1
- g = Recovery Policy 1
- h = Failures generated
- i = Failures generated and/or existing
- j = Malfunction not indicated
- k = Malfunction indicated, diagnose, repair
- l = Undiscovered failures
- m = No undiscovered failures
- n = Recovery Policy 1
- o = Recovery Policy 1

Fig. 9 - Two-Port Network for Example 4

Table 16 - Input Values of Two-Ports Used in Numerical Example 4

Branch	Test											
	1		2		3		4		5		6	
	p	t	p	t	p	t	p	t	p	t	p	t
a	0.8	0	0.90	0	0.99	0	0.85	0	0.85	0	0.99	0
b	0.1	10	0.05	10	0.05	10	0.07	12	0.05	15	0.05	10
c	0.9	5	0.95	1	0.95	1	0.93	3	0.95	10	0.95	1
d	0.8	0	0.90	0	0.90	0	0.85	0	0.80	0	0.90	0
e	0.2	0	0.10	0	0.10	0	0.15	0	0.20	0	0.10	0
f	1.0	2	1.00	2	1.00	2	1.00	2	1.00	3	1.00	1
g	1.0	2	1.00	2	1.00	2	1.00	2	1.00	3	1.00	1
h	0.2	0	0.10	0	0.01	0	0.15	0	0.15	0	0.01	0
i	1.0	0	1.00	0	1.00	0	1.00	0	1.00	0	1.00	0
j	0.1	5	0.05	1	0.05	1	0.09	3	0.05	5	0.01	1
k	0.9	30	0.95	30	0.95	30	0.91	33	0.95	40	0.99	35
l	0.2	0	0.10	0	0.01	0	0.15	0	0.20	0	0.01	0
m	0.8	0	0.90	0	0.99	0	0.85	0	0.80	0	0.99	0
n	1.0	2	1.00	2	1.00	2	1.00	2	1.00	3	1.00	1
o	1.0	2	1.00	2	1.00	2	1.00	2	1.00	3	1.00	1

The output for the six tests is shown in Table 17. Node 1 represents the state of no undiscovered malfunctions at the beginning of the test and node 6 represents the state of no undiscovered malfunctions at the end of the test. Node 11 represents the conditions that there are undiscovered malfunctions at the beginning of a test and node 16 represents the state of undiscovered malfunctions at the end of a test. If the desired goal of analyzing the six tests is to see which was the best one, it is seen in Table 17 that Tests 3 and 6 are better than the other four both from a probability standpoint and from the standpoint of expected time required to perform the test. The distinction between Tests 3 and 6 is not as clear and the choice between

the two would depend on other factors such as the other tests in the countdown, the position of this test in the countdown, and the launch window of the countdown.

Table 17 - GERT Analysis of Two-Ports Described in Table 16.

Branch	p	E(t)	Var (t)	p	E(t)	Var (t)	p	E(t)	Var (t)
	Test 1			Test 2			Test 3		
1-6	.957	18.18	811.2	.991	7.30	291.8	.997	3.38	105.5
1-16	.043	28.11	1111.8	.009	13.85	440.1	.003	13.48	137.0
11-6	.840	57.20	1085.4	.936	42.39	410.6	.947	35.68	115.4
11-16	.160	25.01	1083.0	.064	10.34	409.9	.053	3.62	114.7
	Test 4			Test 5			Test 6		
1-6	.974	12.91	588.1	.986	22.08	943.4	.999	3.66	127.5
1-16	.026	21.97	849.4	.014	30.49	1527.7	.001	13.00	158.7
11-6	.873	53.44	812.4	.925	75.17	1478.9	.989	40.02	140.6
11-16	.127	18.32	810.6	.075	27.01	1476.1	.011	3.96	140.0

BIBLIOGRAPHY ON GERT AND RELATED TOPICS

1. Brock, P. and S. M. Drezner, "A Problem Emanating from a Study of Feedback Loops in Stochastic Networks," The RAND Corporation, P 3276, February, 1966.
2. Deutsch, D., "The Theory of Blocking," Masters Report, Arizona State University, May, 1966.
3. Drezner, S. M. and A. A. B. Pritsker, Network Analysis of Countdown, The RAND Corporation, RM-4976-NASA, March, 1966.
4. Eisner, Howard, "A Generalized Network Approach to the Planning and Scheduling of a Research Program," Journal of Operations Research Society, Vol. 10, No. 1, 1962.
5. Elmaghraby, Salah E., "An Algebra for the Analysis of Generalized Activity Networks," Management Science, Vol. 10, No. 3, April, 1964, pp. 494-514.
6. _____. "On Generalized Activity Networks," Journal of Industrial Engineering, Vol. XVII, No. 11, November, 1966, pp. 621-631.
7. Graham, P., "Profit Probability Analysis of Research and Development Expenditures," The Journal of Industrial Engineering, Vol. XVI, No. 3, May-June, 1965, pp. 186-191.
8. Hopp, W. W., "Flowgraph Techniques for Closed Systems," IEEE Transactions, AES, Vol. 2, May, 1966, pp. 252-264.
9. Hill, T. W., "System Optimization: A Sensitivity Approach Using GERT," unpublished Masters Research Report, Arizona State University, Tempe, Arizona, 1966 (Also in Bulletin #3, Industrial Engineering Research at Arizona State University, 1966).
10. Ishmael, P. C. and A. A. B. Pritsker, User Manual for GERT EXCLUSIVE-OR Program, NASA/ERC, NGR 03-011-034, July 1968.
11. _____. Definitions and Procedures Employed in the GERT EXCLUSIVE-OR Program, NASA/ERC, NGR 03-001-034, July, 1968.
12. Ledbetter, L. R., "A Study of Computational Aspects of Network Models for Planning and Control," Masters Thesis, Lehigh University, 1967.
13. Powell, G.E., "Development, Evaluation and Selection of a Dodge Continuous Sampling Plan When the Rectifying Operation is Not Perfect," Masters Thesis, Lehigh University, 1967.

14. McDonald, W. W., "Analysis of Stochastic Networks," in GRAPH THEORY, Significant Books in Operations Research, Management Information Services, Detroit, Michigan (Also Masters Thesis, Arizona State University, 1966).
15. Pritsker, A. A. B., GERT: Graphical Evaluation and Review Technique, The RAND Corporation, RM-4973-NASA, April, 1966.
16. _____, 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.
17. _____, and G. E. Whitehouse, "GERT: Graphical Evaluation and Review Technique, Part II Probabilistic and Industrial Engineering Applications," Journal of Industrial Engineering, Vol. XVII, No. 6, June, 1966.
18. Smith, R. L., "Stochastic Analysis of Personnel Movement in Formal Organizations," Ph.D. Dissertation, Arizona State University, August, 1967.
19. Werberger, G., "Optimality Considerations of GERT," Masters Thesis, Lehigh University, 1967.
20. Whitehouse, Gary E., "Extensions, New Developments, and Applications of GERT," Ph.D. Dissertation, Arizona State University, August, 1965.