NASr-21(08)

MEMORANDUM
RM-4973-NASA
APRIL 1966

GERT: GRAPHICAL EVALUATION AND REVIEW TECHNIQUE

A. A. B. Pritsker

[redacted]

This research is sponsored by the National Aeronautics and Space Administration under Contract No. NASr-21. This report does not necessarily represent the views of the National Aeronautics and Space Administration.

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA • 90406

PREFACE

This Memorandum, one of a series done by The RAND Corporation on the Apollo Checkout Study for Headquarters, National Aeronautics and Space Administration under Contract NASr-21(08), evolved from a study of the terminal countdown of an Apollo space system. That study, to be reported on separately, developed the concept that a terminal countdown can be represented as a network; this concept in turn created a need for a procedure for (1) analyzing networks that contained activities that had a probability of occurrence associated with them, and (2) treating the plausibility that the time to perform an activity was not a constant, but a random variable. Networks containing these two elements were described by the term "stochastic networks." The result of the research on this problem, presented in this Memorandum, is GERT (Graphical Evaluation and Review Technique), a procedure for the analysis of stochastic networks.

GERT can be a powerful tool for the systems analyst since it has all the advantages associated with networks and provides an exact evaluation of certain types of networks. GERT has wide application possibilities, as indicated by the numerous examples given in this Memorandum, and also has characteristics which make it useful as a teaching mechanism.

The author of this Memorandum is a consultant to The RAND Corporation. His primary association is with Arizona State University, where he is a member of the faculty in the Department of Industrial Engineering.

SUMMARY

GERT, an acronym for Graphical Evaluation and Review Technique, is a procedure for the study of stochastic networks composed of EXCLUSIVE-OR, INCLUSIVE-OR, and AND nodes (vertices) and multiparameter branches (transmittances or edges). In GERT, branches of the network are described by two or more parameters: (1) a probability that the branch is traversed; and (2) the time (or other attribute) to traverse the branch if it is taken. A transformation is developed that combines these two parameters into a single parameter. For EXCLUSIVE-OR nodes, a method is derived for the evaluation of networks in terms of the probability of realizing an output node, and the moment generating function of the time to realize the output node. The total concept of stochastic networks, the transformation, and the evaluation method has been labeled GERT.

For EXCLUSIVE-OR logic nodes, even if the times associated with the branches are random variables, GERT still yields an exact solution. A computer program has been written to obtain such solutions. For the other logic nodes, conceptual and computational problems still exist. These problems are discussed in this Memorandum and approaches and approximations are outlined. As part of the evaluation and review process associated with stochastic networks, a sensitivity analysis of stochastic networks is included with GERT.

In performing the research to derive GERT, it was found that many systems could be described in terms of stochastic networks and that many problems could be solved using GERT. This Memorandum presents the general concepts and fundamentals of GERT. It was decided to

present the research in the manner in which it proceeded because the alternative approach of presenting the transformation derived and then showing that it is an appropriate transformation would tend to lose the organic development which led to the theory.

Throughout the Memorandum, many examples demonstrate the general applicability of GERT. However, GERT as presented is viewed as a starting place from which many avenues of research are possible. This is adequately illustrated in the discussion of future research areas.

The application for which GERT was originally developed--evaluation and review of countdowns--will be discussed in a separate memorandum.

ACKNOWLEDGMENTS

In the formulation stages, Dr. W. W. Happ of Arizona State University contributed ideas and concepts which were instrumental in the development of the theory of GERT. Mr. Stephen M. Drezner of The RAND Corporation and Mr. Gary E. Whitehouse of Arizona State University gave significant assistance to the author through discussions of the development and application of GERT.

The program presented in Appendix A was formulated and developed by Mr. Alfred B. Nelson of The RAND Corporation.

APOLLO CHECKOUT SYSTEM STUDY BIBLIOGRAPHY

RM-3827-NASA Johnson, R. E., and C. T. Whitehead, A Preliminary Study of the Apollo In-flight Test System, November 1963.

RM-3888-NASA Brom, J. R., J. J. Ferrante, S. I. Firstman, L. T. Mast, and D. S. Stoller, Objectives, Measures of Effectiveness, and Criteria for Apollo Prelaunch Operations as They Affect Checkout Activities, October 1963.

RM-3934-NASA Firstman, S. I., Operational Methods for Space Vehicle Countdown Planning, November 1963.

RM-4078-NASA Brom, J. R., D. J. Dugas, E. M. Scheuer, and C. G. Vandervoort, On-board Checkout and Remedial Activities for the Lunar Excursion Module (U), May 1964, Confidential.

RM-4095-NASA A. F. Cahn, A Simulation for Sensitivity Testing of Various Parameters Involved in Apollo Flights, September 1964.

RM-4096-NASA Kristy, N. F., Apollo Astronaut-Crew Skill and Training, June 1964, FOUO.

RM-4111-NASA Dugas, D. J., The Lunar Environment: Implications for Apollo Crew Operations (U), August 1964, Confidential.

RM-4150-NASA Stoller, D. S., and K. Harris, Evaluation Model of the Effectiveness of the Countdown-Checkout Phase of Apollo Pre-launch Operations, July 1964.

RM-4160-NASA Mast, L. T., L. D. Amdahl, O. T. Gatto, and A. A. B. Pritsker, Growth of Automation in Apollo Prelaunch Checkout: Volume I, Go/No-Go Testing, July 1964.

RM-4161-NASA Mast, L. T., L. D. Amdahl, O. T. Gatto, and A. A. B. Pritsker, Growth of Automation in Apollo Prelaunch Checkout: Volume II, Added Roles, July 1964.

RM-4177-NASA Whitehead, C. T., Selection of Spares and Redundancy for the Apollo Spacecraft, August 1964.

RM-4200-NASA Pepler, R. D., and J. G. Wohl, Dunlap and Associates, Inc., Display Requirements for Prelaunch Checkout of Advanced Space Vehicles, June 1964.

RM-4214-NASA Dordick, H. S., In-flight LEM Checkout Prior to Lunar Operations (U), September 1964, Confidential.

RM-4272-NASA Truelove, A. J., Scheduling of Apollo In-flight Checkout, September 1964.

RM-4311-NASA Kristy, N. F. and H. P. Roth, Astronaut-Crew Requirements for Apollo In-flight Operations, September 1964.

RM-4325-NASA Vandervoort, C. G., and H. S. Dordick, Selecting Between Redundancy and Repair in Manned Spacecraft: A Designer's Tool, September 1964.

RM-4455-NASA Amdahl, L. D., and L. T. Mast, A Proposed Method for Onboard Monitoring of Discrete Signals, February 1965.

RM-4506-NASA Pepler, R. D., and J. G. Wohl, Dunlap and Associates, Inc., Integration of Man and Computer in Prelaunch Checkout of Advanced Space Vehicles, April 1965.

RM-4518-NASA Scheuer, E. M., Two Models for Time Constrained Maintenance, April 1965.

RM-4565-NASA Drezner, S. M., and O. T. Gatto, Computer-Assisted Countdown: Preliminary Report on a Test of Early Capability, May 1965.

RM-4581-NASA Brooks, R. B., and R. H. Haase, A Network Approach to Parts Provisioning for Apollo Prelaunch Operations, September 1965.

RM-4678-NASA Chesler, L. H., and R. Turn, The Monitoring Task in Automated Checkout of Space Vehicles, September 1965.

CONTENTS

I. INTRODUCTION

Networks and network analyses are playing an increasingly important role in the description and improvement of operational systems primarily because of the ease with which systems can be modeled in network form. This growth in the use of networks can be attributed to: (1) the ability to model complex systems by compounding simple systems; (2) the mechanistic procedure for obtaining system figure-of-merits from networks; (3) the need for a communication mechanism to discuss the operational system in terms of its significant features; (4) a means for specifying the data requirements for analysis of the system; and (5) to provide a starting point for analysis and scheduling of the operational system. This last reason was the original reason for network construction and use. The advantages that accrued outside of the analysis procedure soon justified the network approach; however, further efforts toward improving and extending network analysis procedures have not kept pace with the applications of networks.

In this Memorandum a new procedure for analyzing networks with stochastic and logical properties is developed. This procedure makes it possible to analyze complex systems and problems in a less inductive manner and hence should stimulate efforts in the network analysis area. Although the research reported solves just one problem among many in the network field, it provides a breakthrough which should simplify the development of analysis procedures for more complex type networks. The name given to the technique developed is GERT, Graphical Evaluation and Review Technique.

GERT is a technique for the analysis of a class of networks which have the following characteristics: (1) a probability that a branch of the network is indeed part of a realization of the network; and (2) an elapsed time* or time interval associated with the branch if the branch is part of the realization of the network. Such networks will be referred to as stochastic networks and consist of a set of branches and nodes. A realization of a network is a particular set of branches and nodes which describe the network for one experiment. If the time associated with a branch is a random variable, then a realization also implies that a fixed time has been selected for each branch. GERT will derive both the probability that a node is realized and the conditional moment generating function (M.G.F.) of the elapsed time required to traverse between any two nodes.

A note of caution is necessary at this point to forewarn the reader about terminology. Since GERT deals with a composite-type graph, there will be some terminology that differs from the standard network and the standard signal flowgraph terminologies which already differ. Two illustrations of the terminology differences are:

1. The term branch is used throughout to indicate an activity between two nodes (milestones). In GERT a branch always has a direction. In signal flowgraphs the word transmittance is used in this connection. The value of the transmittance is a <u>parameter</u> of the system. In GERT a branch can have multiple values associated with it, some of which can be random variables. The use of the statistical term "random variable" to describe a quantity associated with a branch is called to the attention of the reader.

---

*In this report, time is used in the generic sense to represent a variable that is additive in the sense to be described below. As an extension, variables that are multiplicative will be considered in Appendix C.

2. In GERT, there are probabilities associated with each branch. These probabilities represent the relative frequency* that the branch is part of the network. When the branch is part of the network it is said that the branch is realized. Since branches lead to nodes, the concept of realization also pertains to nodes.

The above two illustrations are cited in the hope that the reader will recognize the difference inherent in GERT and adapt to its terminology.

Section II presents the components of stochastic networks, along with an example to illustrate the integration of the components into a network. Then the steps in GERT are discussed and the analysis problems inherent in GERT presented. At the end of this section, past research related to GERT is outlined.

Section III sets forth the derivation of the equivalent network (a multibranched network reduced to a one-branch network) for three basic networks: series; parallel; and self-loop, and makes a generalization of the derivation procedure based on flowgraph theory. In the next three sections, the procedures discussed in Sec. III are illustrated and applied.

Concepts of confidence statements, sensitivity, and elasticity are presented in Sec. VII. In Sec. VIII, a summary list of areas for future research is given.

Appendix A presents a digital computer program for analyzing specific GERT networks. Appendix B discusses in greater detail the AND and INCLUSIVE-OR nodes. While the development of GERT in the text

---

*The use of subjective probabilities associated with a branch is similar to the use of subjective probabilities in other areas and the same rules apply to their use in GERT.

of this Memorandum was restricted to additive parameters, Appendix C rounds out the exposition by discussing stochastic networks with multiplicative parameters.

II. COMPONENTS OF STOCHASTIC NETWORKS

The components of stochastic networks are directed branches (arcs, edges, transmittances) and logical nodes (vertices). A directed branch has associated with it one node from which it emanates and one node at which it terminates. Two parameters are associated with a branch: (1) the probability that a branch is taken, p_a, given that the node from which it emanated is realized; and (2) a time, t_a, required, if the branch is taken, to accomplish the activity which the branch represents.* t_a can be a random variable. If the branch is not part of the realization of the network then the time for the activity represented by the branch is zero. The visual representation of a directed branch, without the nodes represented, is:

$$\xrightarrow{(p_a;t_a)}$$

A node in a stochastic network consists of an input (receiving, contributive) side and an output (emitting, distributive) side. In this Memorandum three logical relations on the input side and two types of relations on the output side will be considered.** The three logical relations on the input side are:

---

*Generalization to permit more than one additive parameter presents no conceptual difficulty as long as the parameters are independent (see Example 12, p. 62).

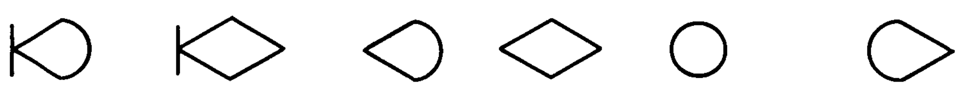
**In Appendix B, two other logical nodes, a minimum node and an inverter node, are proposed.

Name	Symbol	Characteristic
EXCLUSIVE-OR		The realization of any branch leading into the node causes the node to be realized; however, one and only one of the branches leading into this node can be realized at a given time.
INCLUSIVE-OR		The realization of any branch leading into the node causes the node to be realized. The time of realization is the smallest of the completion times of the activities leading into the INCLUSIVE-OR node.
AND		The node will be realized only if all the branches leading into the node are realized. The time of realization thus is the largest of the completion times of the activities leading into the AND node.

On the output side, the two relations are defined as:

Name	Symbol	Characteristic
DETERMINISTIC		All branches emanating from the node are taken if the node is realized, i.e., all branches emanating from this node have a p-parameter equal to 1.
PROBABILISTIC		Exactly one branch emanating from the node is taken if the node is realized.

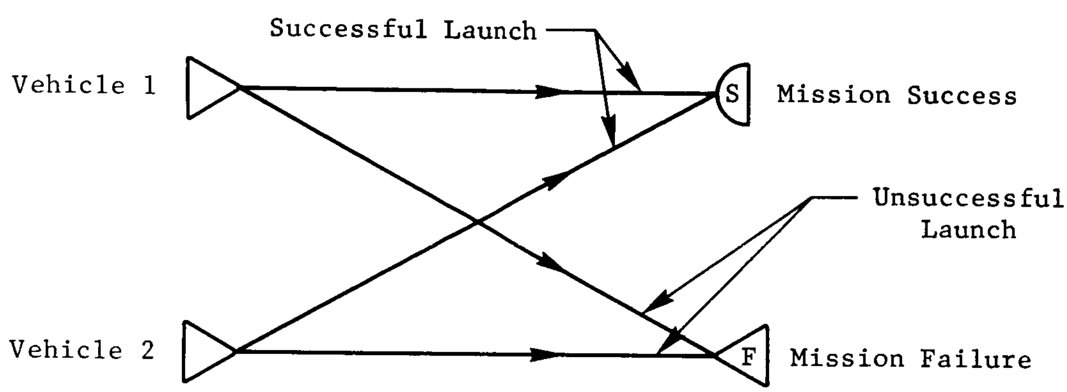
For notational convenience, the input and output symbols are combined below to show that there are six possible types of nodes:



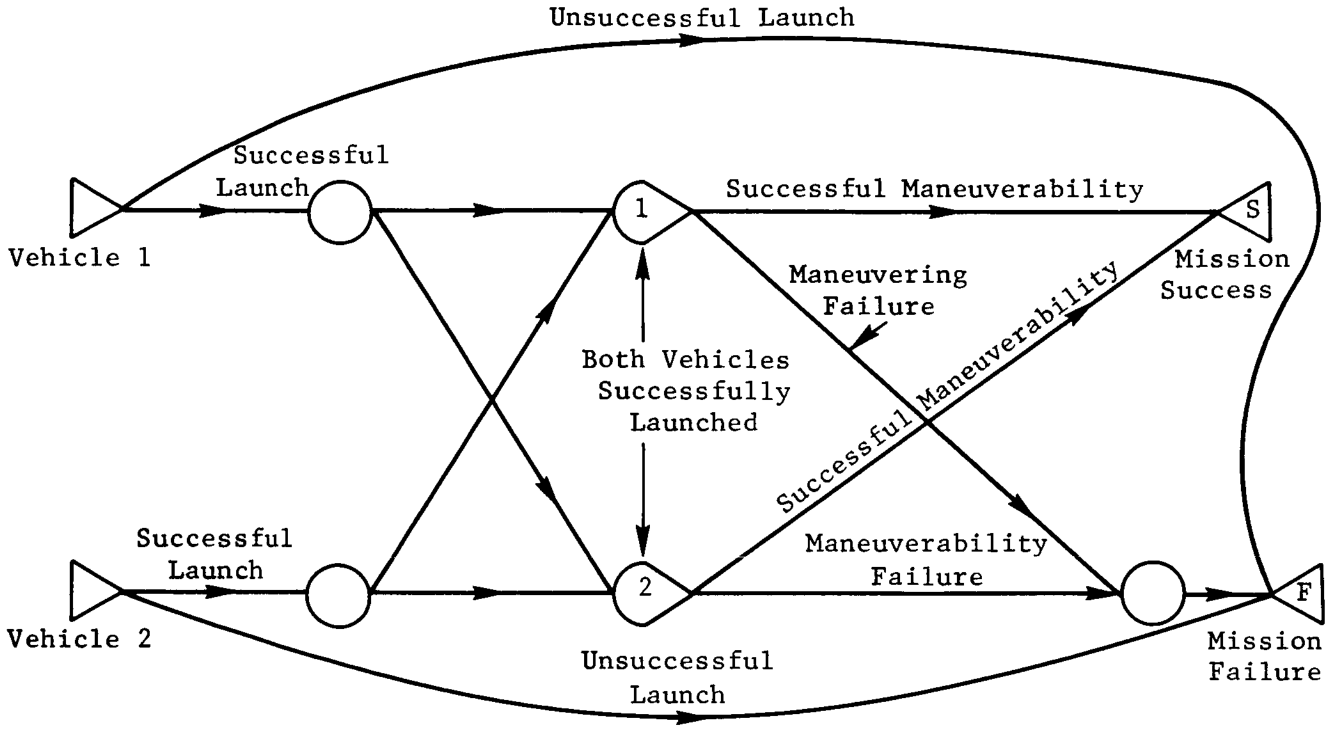
Before proceeding to the mathematical analysis of these different node types, an example of the use of stochastic networks in modeling is given to assist in understanding stochastic networks. The example is for illustrative purposes only, and no analysis will be performed on the derived networks.

EXAMPLE OF MODELING USING STOCHASTIC NETWORKS

Consider a space mission involving the rendezvous of two vehicles. In order for the mission to have a chance for success, both vehicles must be successfully launched. The stochastic network for this problem is:

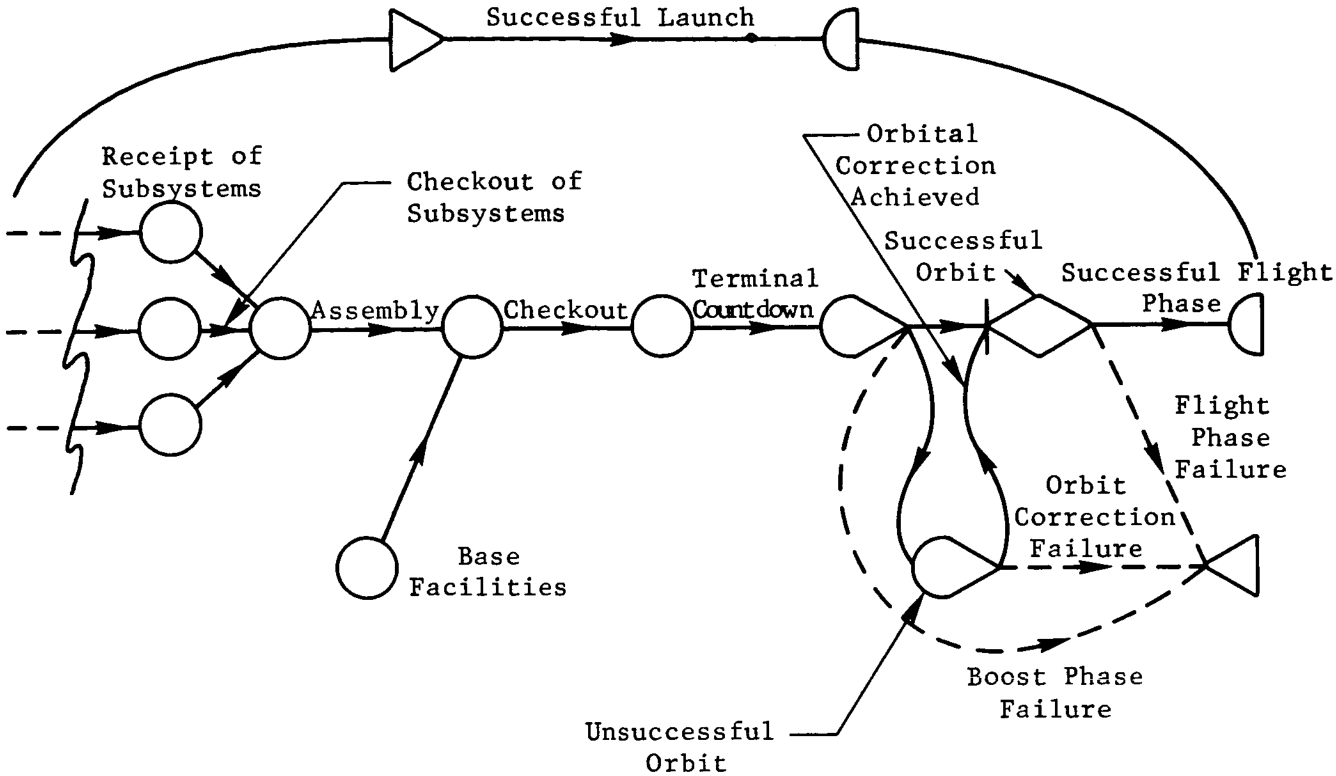


For the node S to be realized, both branches leading into it must be realized (a characteristic of the AND node). Node F will be realized if either branch incident to it is realized (INCLUSIVE-OR node). Obviously the above model is simple, but it does illustrate the modeling and communication aspects of stochastic networks. To extend the model somewhat, assume that if both vehicles are successfully launched, at least one of the vehicles must be capable of maneuvering for the mission to be a success. The network for this situation is:



In this case nodes 1 and 2 are added to specify the event both vehicles successfully launched. The S node now will be realized if either branch incident to it is realized since the assumption was that a maneuverability capability was necessary only for one vehicle to obtain mission success.

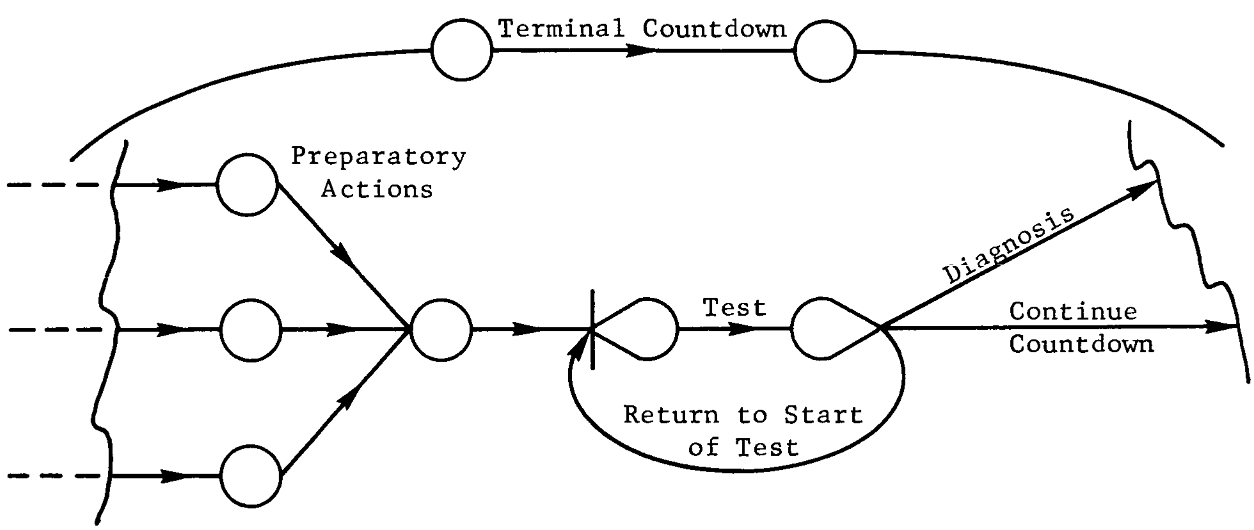
The above networks represent highly aggregated models of complex operations. One of the beauties of stochastic networks is its usefulness at many levels within a problem area. For example, the branch "successful launch" can be divided into many branches and nodes. The following illustrates this concept:



In this network it is seen that the AND node plays a predominant role in the activities up to and including the terminal countdown. This is due to the fact that all activities must be performed prior to lift-off. This, of course, is a simplified view of the system; however, it serves the purpose of illustrating that part of a stochastic network can be a PERT-type network. After the terminal countdown, either-or possibilities are presented and the probabilistic output node is shown. The event represented by the node labeled "successful orbit" is an EXCLUSIVE-OR node since a successful orbit can occur in two mutually exclusive ways: (1) proper operation during boost phase, and (2) unsuccessful orbit after boost phase with orbit correction achieved. The dotted lines represent activities that do not contribute to the "successful launch" but are branches associated with the system modeled.

In this case they would lead to the node "unsuccessful launch" which is an INCLUSIVE-OR node because any of the branches leading into the node can be realized and any of them causes the node to be realized.

Continuing the example, consider the branch "terminal countdown," a segment of which can be represented as below:



The network shows three preparatory actions to a test, such as power-on, stimuli calibrated, and recorder-on, which are required before the test can begin. The test is performed and, based on the results of the test, the countdown is continued, diagnosis is initiated, or the test is performed over. This last action illustrates the concept of feedback in a stochastic network.

Obviously the above are not complete descriptions, but they illustrate the communication capabilities of GERT. Also, by decomposing the problem into segments, the parameters of interest for an aggregate model can be computed. Thus the probability of a successful launch could be computed by evaluation of the more detailed networks.

STEPS IN APPLYING GERT

The foregoing material described the qualitative aspects of GERT. Basically, the steps employed in applying GERT are:

1. Convert a qualitative description of a system or problem to a model in network form;

2. Collect the necessary data to describe the branches of the network;

3. Obtain an equivalent one-branch function between two nodes of the network;

4. Convêrt the equivalent function into the following two performance measures of the network:

    a. The probability that a specific node is realized; and

    b. The M.G.F. of the time associated with an equivalent network;

5. Make inferences concerning the system under study from the information obtained in 4 above.

In this expository Memorandum emphasis will be directed toward items 3 and 4. Discussion of item 1 will be given through examples. For items 2 and 5, the methods employed in PERT and flowgraph theory are equally applicable for GERT.

BASIC NETWORK ANALYSIS

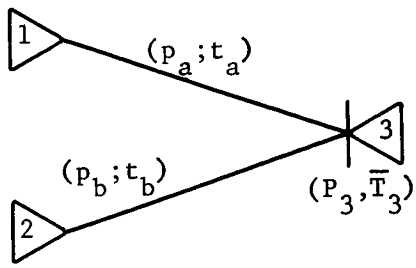
The basic algebra associated with the previously defined nodes is set forth below. Although analyzing networks through the development of a technique that includes all six types of nodes appears formidable, there are several saving facts.

First, all six nodes behave in the same manner if only one branch is received at the input side and one branch is emitted on the output side. Thus, if only two branches are under consideration and they are in series, the node type has no effect on the equivalent one-branch network. An equivalent network is defined as a reduction of a multibranched network into a one-branch network, where the parameters of the one-branch network are derived from the parameters of the branches of the multibranched network.

Second, the concept of feedback is only appropriate for the EXCLUSIVE-OR input type of node. This results from the fact that feedback requires that the node being returned to be realized prior to the feedback. But the node cannot be realized if it is an AND type node unless all inputs have been realized. For the INCLUSIVE-OR input type, only the branch representing the first activity completed is significant. All other branches are ignored in computing the time the INCLUSIVE-OR node is realized. Since a fedback branch will always be completed after a non-feedback branch, the EXCLUSIVE-OR representation can replace the INCLUSIVE-OR node if a feedback branch is incident to the node.

Third, if all the nodes have the EXCLUSIVE-OR input characteristics, then either all node outputs are of the probabilistic type, or the paths (collections of branches) following a deterministic output are independent (nontouching, disjoint). If this were not the case then at some input side of a node there would be a possibility of two branches being realized simultaneously, which contradicts the condition that all nodes of the network have the EXCLUSIVE-OR input relation.

Fourth, for some networks AND and INCLUSIVE-OR input types can be converted to the EXCLUSIVE-OR relationship. To illustrate this, each of these relationships is discussed in a quantitative fashion below. For the EXCLUSIVE-OR relation we have



$$P_3 = P_1p_a + P_2p_b$$

$$\text{and } \bar{T}_3 = \frac{P_1p_a(\bar{T}_1 + t_a) + P_2p_b(\bar{T}_2 + t_b)}{P_1p_a + P_2p_b}$$

where P_i is the probability that node i is realized, and $\bar{T}_i$ is the expected time that node i is realized, given that it is realized. For this introductory discussion, only the expected time for a node to be realized, given it is realized, will be calculated. (Note that even though t_a and t_b may be constants, the time to realize node 3, T_3, is a random variable.) The derivation of P_3 and $\bar{T}_3$ is by enumeration of the possible events that result in the realization of node 3. Node 3 can be realized if either branch a or branch b is realized. The probability that branch a will be realized is the probability that node 1 is realized, P_1, times the probability that branch a is realized given node 1 is realized, which is p_a. A similar discussion holds for branch b and the equation for P_3 results. Note by definition of the EXCLUSIVE-OR relation, branches a and b cannot both occur. If this were a possibility, then node 3 would have to be an INCLUSIVE-OR node. The expected time to realize node 3, given it is realized, is the weighted sum of the possible times to realize node 3.

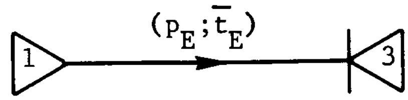
If node 1 were the same as node 2, then $P_1 = P_2$ and $\bar{T}_1 = \bar{T}_2$, and the following equations result:

$$P_3 = P_1(p_a + p_b)$$

and

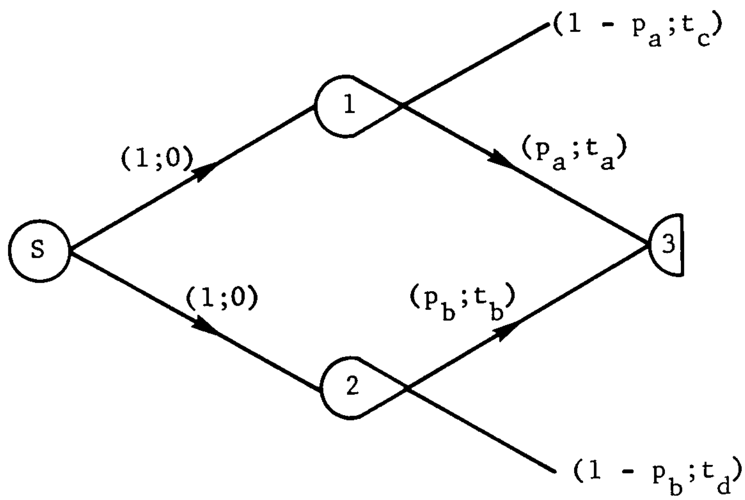
$$\bar{T}_3 = T_1 + \frac{p_a t_a + p_b t_b}{p_a + p_b} ,$$

and the network could be drawn as



where $\qquad p_E = p_a + p_b \quad$ and $\quad \bar{t}_E = \dfrac{p_a t_a + p_b t_b}{p_a + p_b}$.

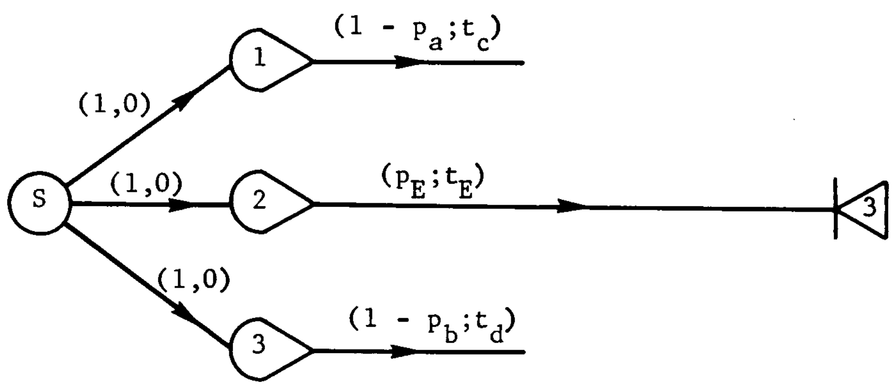
Consider next the AND logic relation as depicted below:



Node 3 will only be realized if both a and b are realized. The probability that a is realized is $P_1 p_a$ and the probability that b is realized is $P_2 p_b$. The probability that both are realized is the intersection of $P_1 p_a$ and $P_2 p_b$. In this case the intersection of the events associated with nodes 1 and 2, denoted by $P_{1\cap 2}$, is equal to P_1, and assuming $p_{a\cap b}$ is $p_a p_b$, we have $P_3 = P_1 p_a p_b$. Since both branches must be realized, we have

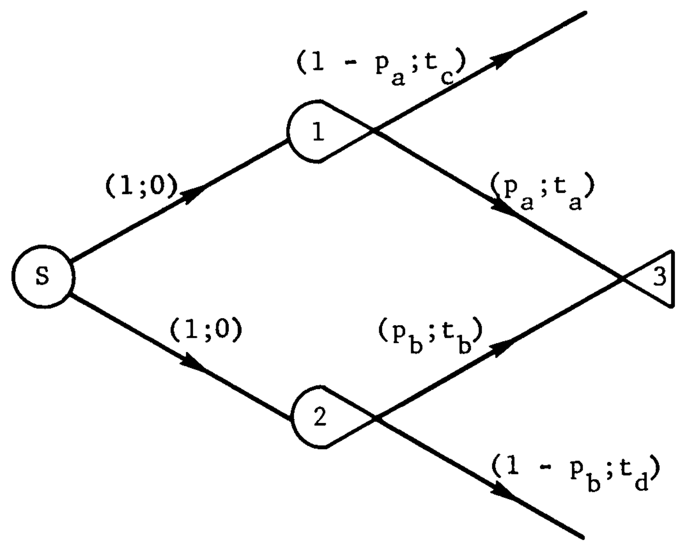
$$T_3 = \max\,(T_1 + t_a;\ T_2 + t_b)\ .$$

Care must be taken here in the computation of expected values since the expected value of a maximum is not usually the maximum of the expected values. This will be discussed in Appendix B. For this case $T_1 = T_2 = T_S$, and we have $T_3 = T_S + \max\,(t_a;t_b)$. Thus $p_E = p_a p_b$ and $t_E = \max\,(t_a;t_b)$, and the equivalent network would be

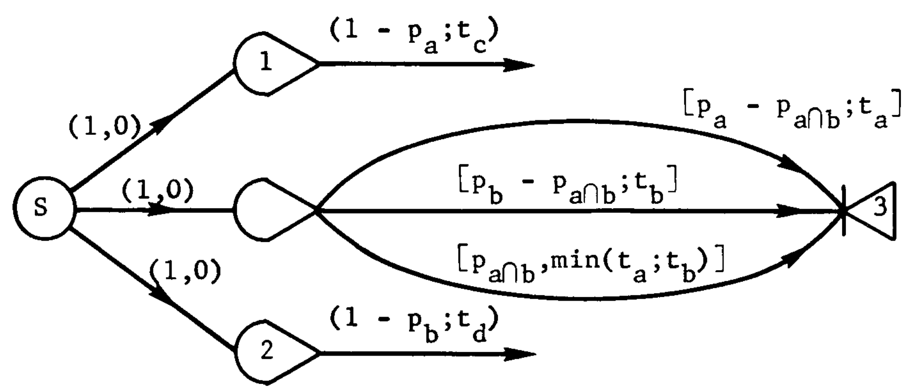


The EXCLUSIVE-OR relation can replace the AND relation at node 3 since only one branch is received at node 3.

For the INCLUSIVE-OR relation, the analysis proceeds as in the AND case. The branches of the network given below



are equivalent to



The reduction process involves the enumeration of all mutually exclusive alternative methods of realizing node 3 from node S. These are:

Description	Probability	Equivalent time
Branch a but not branch b	$p_a - p_{a\cap b}$	t_a
Branch b but not branch a	$p_b - p_{a\cap b}$	t_b
Branch a and branch b	$p_{a\cap b}$	min $(t_a;t_b)$

These examples demonstrate the complexity of converting input relations for simple networks. However, the EXCLUSIVE-OR relationship appears basic.

SCOPE OF THIS MEMORANDUM

GERT is a procedure for studying stochastic networks. It provides a framework on which future research on stochastic networks can be performed. The main body of this Memorandum is devoted to the analysis and application of stochastic networks consisting only of EXCLUSIVE-OR nodes. This has been selected as the starting place for two reasons: (1) the EXCLUSIVE-OR node represents a linear type operator and, hence, is the easiest node type to analyze; and (2) the EXCLUSIVE-OR node is a fundamental element of stochastic networks, as discussed in the previous subsection.

The research reported herein will provide insights into the analyses of other logical nodes. The AND node is discussed at length in Appendix B so as to foster further research in that area. In fact, throughout this Memorandum areas within GERT requiring further research are indicated. Thus the scope of GERT is broad. The scope of this Memorandum, however, is limited to the presentation of two basic tools: (1) a framework for the study of stochastic networks; and (2) an exact procedure for analyzing networks with only EXCLUSIVE-OR nodes (along with a computer program for performing the necessary calculations).

RELATED RESEARCH

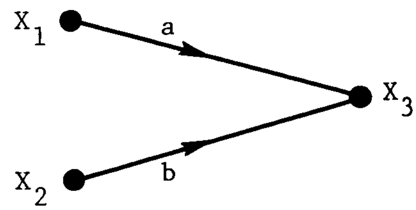
Research related to GERT has been concerned with project-scheduling-type networks (PERT, CPM variety) and signal flowgraphs. For PERT-type networks, all branches must be taken; hence, a realization of the network is the entire network. A time (or distribution of times) is associated with each branch of the PERT network. An analysis is performed to determine the distribution of the total project time. This analysis requires rather severe assumptions, and only an approximation to the distribution of the total project time is obtained.[1-4]

Eisner[5] suggested the use of logical elements in the PERT-type networks, and Elmaghraby[6] developed a notation for a multiparameter branch network and the logical elements previously presented. Elmaghraby also developed an algebra and coined the phrase "generalized activity networks" to describe such networks. Elmaghraby's algebra is limited to branches that have constant times associated with them.

The research in the area of flowgraphs has been extensive. Two surveys of the techniques by Lorens[7] and Happ[8] give a total of over 200 references. A topological presentation is given by Kim and Chien.[9]

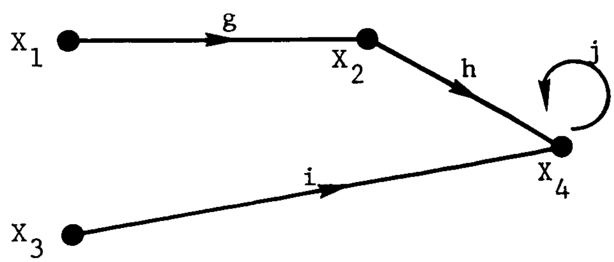
A flowgraph consists of transmittances (directed branches) and nodes. For the usual flowgraph, a realization of the graph must contain all the transmittances of the graph. Huggins[10] and Howard[11,12] have employed flowgraphs to represent and analyze probabilistic systems.

A basic property of flowgraphs is the law of nodes, i.e., the value of a variable associated with a node of the graph is equal to the sum of the values associated with transmittances terminating at that node times the value of the node from whence the transmittance originated. Two examples of this law are:



$$X_3 = aX_1 + bX_2$$

and



$$X_4 = hX_2 + jX_4 + iX_3$$

$$X_4 - jX_4 = h(gX_1) + iX_3$$

$$X_4 = \frac{hgX_1 + iX_3}{1 - j} .$$

With this property it can be shown that a flowgraph is a convenient graphical representation of a set of independent simultaneous linear equations. Thus for the above flowgraph there is an equation for each node excluding the inputs nodes, viz.,

$$\begin{aligned} - gX_1 + X_2 \qquad\qquad\qquad\qquad &= 0 \\ - hX_2 - iX_3 + X_4 - jX_4 &= 0 \end{aligned}$$

The rule for constructing the equations is to subtract from the node value the sum of the product of the transmittances entering the nodes times the value of the node from whence the transmittance originated.

III. STOCHASTIC NETWORKS WITH EXCLUSIVE-OR NODES

In this section the equivalent network will be derived for the following three basic networks: (1) series; (2) parallel; and (3) self-loop. The derivation will be accomplished by enumerating all possible paths from the starting node (source) of the network to the terminal node (sink) of the network. A generalization of the derivation procedure based on flowgraph theory will then be made. The generalization permits the analysis of networks where branches represent activities having durations described by random variables. To simplify the terminology, a node will be described by its input relationships, where no ambiguity is thought to be present.

Figure 1 illustrates the equivalent one-branch network for a series, a parallel, and a self-loop network, all of whose branches have constant time parameters and whose nodes are of the EXCLUSIVE-OR type.

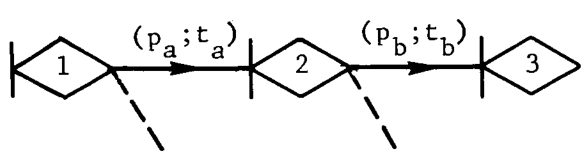
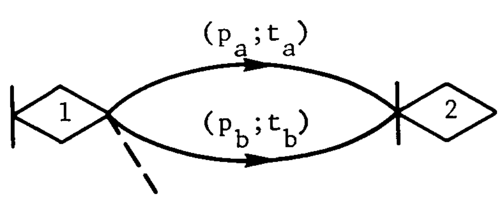
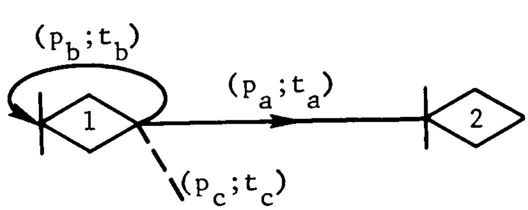
Network type	Representation with constant times	Equivalent probability	Equivalent expected time
(a) Series		$p_a p_b$	$t_a + t_b$
(b) Parallel		$p_a + p_b$	$\frac{p_a t_a + p_b t_b}{p_a + p_b}$
(c) Self-loop		$\frac{p_a}{1 - p_b}$	$t_a + \left(\frac{p_b}{1 - p_b}\right) t_b$

Fig. 1--Equivalence calculations of basic networks

Since for a series network both branches must be taken to reach node 3, the probability of taking both branches is the product of the individual probabilities and the time is the sum of the individual times. For the parallel branches, either branch can be part of the realization but not both (by definition of the "EXCLUSIVE-OR" element). Thus the probability of going from node 1 to node 2 is the sum of the probabilities. The time to traverse from node 1 to node 2 is no longer a constant but takes on the value t_a with probability p_a, and t_b with probability p_b. Thus the equivalent time to realize node 2, given that it is realized, is a random variable. Normalizing p_a and p_b by dividing each by $(p_a + p_b)$ to ensure that the complete density function for the equivalent time is accounted for, we have the equivalent expected time, as shown in part (b) of Fig. 1. It should be clear that a complete description of the time to realize node 2 has not been obtained, and the use of the expected value to describe the time parameter is an approximation.

Reduction of the self-loop to an equivalent probability and an equivalent expected time is obtained by summation of the probabilities and probable times of all possible paths from node 1 to node 2. The probability of going from 1 to 2 with no transitions around the self-loop is p_a; with one transition around the self-loop is $p_a p_b$; with n transitions it is $p_a p_b^{\,n}$. Summing yields $p_E = \dfrac{p_a}{1 - p_b}$. Similarly

$$E\{t\} = \sum_{n=o}^{\infty} [nt_b + t_a] \frac{p_a p_b^{\,n}}{p_a/(1 - p_b)} = t_a + \left[\frac{p_b}{1 - p_b}\right] t_b \ ,$$

where the normalizing factor is $p_a/(1 - p_b)$. Note that the parameters of the c-branch must also be altered by the same factors if the self-loop is removed from the network. Again the expected time does not completely describe the network.

From the analysis of the basic networks presented above, it is seen that for two branches in series the probabilities associated with the branches are multiplied to obtain the equivalent probability for the two branches. For parallel branches, the probabilities add. These rules adhere to the basic law of nodes presented previously for flowgraphs, i.e., the probability associated with a node can be computed as the sum of the probabilities of each incoming branch times the probability of the node from which the branch emanated. Thus if time was not associated with the network, the network analysis could be accomplished using flowgraph theory. Alternatively, by setting all times on a stochastic network to zero and allowing the other parameter (probability) to assume a wider range of values reduces a stochastic network to a flowgraph.

It is now possible to state the relationship between PERT-type networks and flowgraphs and stochastic networks:

1. PERT-type networks are stochastic (GERT-type) networks with all AND-deterministic nodes.

2. Flowgraphs are stochastic networks with a single multiplicative parameter (all additive parameters such as time are set to zero). The probabilistic interpretation for the multiplicative parameter is removed.

Returning to the discussion of the reduction of the basic networks, it is seen that the time parameter is added for two branches in series and is a weighted average for two branches in parallel. These observations suggest the transformation of p and t into a single function, $w(s) = pe^{st}$. Then for two branches in series, the w-function of the branches will be multiplied, e.g., $w_E(s) = w_1(s)\ w_2(s)$ and for two branches in parallel the w-functions of the branches will be added, e.g., $w_E(s) = w_1(s) + w_2(s)$. Differentiating with respect to s and then setting s = 0 yields a result proportional to the expected times. In the next two subsections the technique for using this transformation within GERT for analyzing stochastic networks is presented in detail.

NETWORK ANALYSIS EMPLOYING A TOPOLOGICAL EQUATION

In the preceding paragraph, the w-function was suggested as a transformation device. There are two reasons for this transformation:

1. the parameters of the stochastic network are combined in the desired fashion; and

2. the w-function obeys the law of nodes of flowgraph theory and, hence, the topological equation of flowgraph theory can be employed to analyze stochastic networks.

The method for obtaining the desired information from the equivalent w-function will now be discussed. Since the time parameter does not affect the equivalent probability, the equivalent probability can be obtained by setting the dummy variable s equal to zero. Thus

$$p_E = w_E(0) \ . \tag{1}$$

For two branches in series, $w_E(s) = w_1(s)w_2(s) = (p_1e^{st_1})(p_2e^{st_2})$ and hence, $p_E = w_E(0) = p_1p_2$, as desired. For two branches in parallel, $w_E(s) = w_1(s) + w_2(s) = p_1e^{st_1} + p_2e^{st_2}$ and $p_E = w_E(0) = p_1 + p_2$, as desired. For the equivalent time, it is seen that by differentiation of $w_E(s)$ with respect to s and then setting s = 0, an expression proportional to the expected time results, viz., for two branches in series $\left.\frac{\partial w_E(s)}{\partial s}\right|_{s=0} = p_1p_2(t_1 + t_2)$ and for two branches in parallel

$\frac{\partial w_E(s)}{\partial s}\Big|_{s=0} = p_1t_1 + p_2t_2$. For both of these expressions the division by p_E will yield the desired results for the equivalent expected time. The need for this division is due to the fact that the equivalent time is a conditional variable, i.e., conditioned on the branch being realized. From the above, it is seen that

$$\mu_{1E} = \frac{\partial}{\partial s}\left[\frac{w_E(s)}{w_E(0)}\right]_{s=0}$$

where μ_{nE} is defined as the n^{th} moment about zero of the equivalent branch.

Further exploration shows that

$$\mu_{nE} = \frac{\partial^n}{\partial s^n}\left[\frac{w_E(s)}{w_E(0)}\right]_{s=0} \qquad (2)$$

and, hence, $\frac{w_E(s)}{w_E(0)} = M_E(s)$ is the M.G.F.* of the equivalent time, t_E. It is convenient at this time to define the n^{th} cumulant, K_{nE} (for $n \leq 3$ the cumulants yield the moments about the mean directly), which is given by

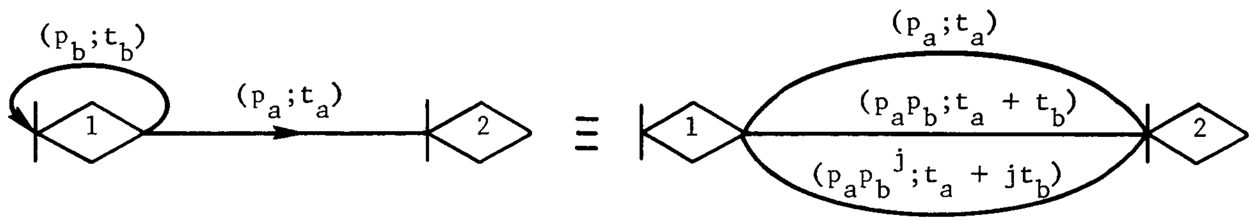
$$K_{nE} = \frac{\partial^n}{\partial s^n}\left[\ell n\ M_E(s)\right]_{s=0} \quad . \qquad (3)$$

Thus, the second moment about the mean, the variance, can be obtained directly as K_{2E}. Equations 1, 2, and 3 hold for all branches with the subscript E replaced by the subscript of the branch under consideration.

The w-function was developed based only on the series and parallel basic networks. However, it can be shown that any network is a combina-

*Due to the definition of the time parameter, t_E is the equivalent time, given that the equivalent network is realized. Thus $M_E(s)$ is the M.G.F. of the conditioned equivalent time parameter.

tion of series and parallel equivalent networks. The self-loop is a good example. It consists (as is illustrated below) of an infinite number of equivalent branches in parallel with each branch having a probability associated with it of $p_a p_b^j$ and a time of $t_a + jt_b$, *viz.*,



For the j^{th} branch, there are j branches equivalent to the feedback branch b in series with the forward branch a. This example illustrates the point that if the w-function holds for both the series and parallel networks, it will hold for a network of arbitrary complexity.

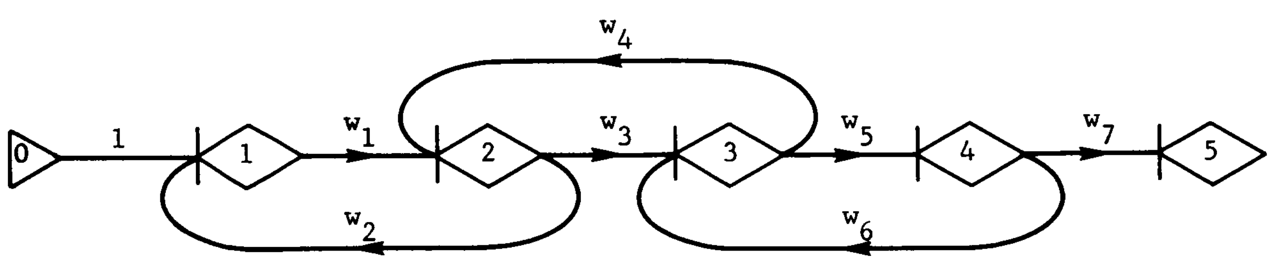
In order to employ w-functions effectively, it is necessary to derive a procedure for obtaining $w_E(s)$ from knowledge of the $w_j(s)$ functions for the individual branches j. A systematic approach to the evaluation of systems of arbitrary complexity is provided by the topological equation of signal flowgraph theory. The topological equation holds for independent linear systems of equations and it specifies the value of the determinant of the matrix of coefficients of the equations. The transformation of p and t into the w-function combines the variables of interest into a linear form. This is seen from the correspondence between the law of nodes and a set of linear equations discussed on p. 19. Thus the w-function satisfies the conditions necessary for its use in the topological equation.

The topological equation describes the relationship between branches $w_j(s)$ for any network. The topological equation[13] is

$$H(s) = 1 + \sum_m \sum_i (-1)^m L_i(m) \quad , \tag{4}$$

where $L_i(m)$ is the loop product of m disjoint (nontouching) loops (m = 1, 2, 3, ..., is called the order of the loop) and the summation is overall combinations of the m disjoint loops. A loop is defined as a sequence of branches such that every node is common to two and only two branches of the loop, one terminating at the node and the other emanating from that node. In a first-order loop every node can be reached from every other node. A loop of order n is a set of n disjoint first-order loops. Disjoint loops are loops which have no nodes in common. The parameter of a loop is the product of the parameters of the branches of the loop.

Before proceeding with the use of the topological equation, an example of the concept of a loop will be given. Consider the network shown below (suppressing the s for convenience):



By the definition given of a loop, it is seen there are three loops of order 1: $L_1(1) = w_1w_2$; $L_2(1) = w_3w_4$; and $L_3(1) = w_5w_6$. The reason these are loops is that the nodes of the loop are such that each branch of the loop enters and leaves only one node of the loop. The w-function of the loop is the product of the w-functions of the branches of the loop (as if the branches were in series). Now loops $L_1(1)$ and $L_3(1)$ do not have a node in common, therefore they are disjoint (nontouching) and are combined to form a loop of order 2, i.e.,

$$L_1(2) = L_1(1)L_3(1) = w_1w_2w_5w_6 \; .$$

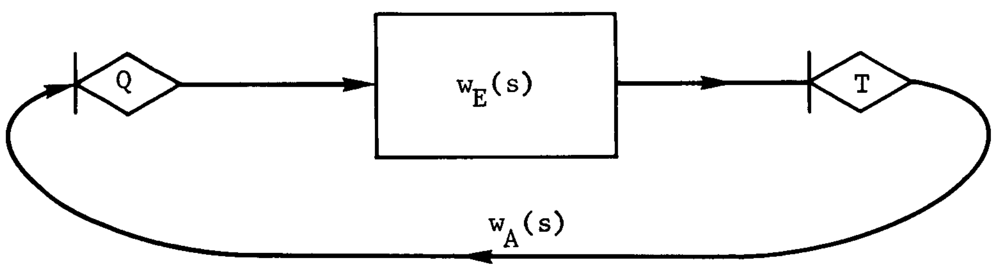
Another concept of importance in flowgraph theory is that of the forward path. Given two nodes, a forward path is a sequence of branches from one node to the other such that every node except the two specified is common to two and only two branches of the forward path. The difference between a forward path and a loop is that the start node has no input branch and the terminal node has no output branch. For the above network, if node 1 is defined as the start node and node 5 as the terminal node, then there is only one forward path, which is $w_1w_3w_5w_7$. The node 0 is included in the network to comply with the definition of a forward path. This node will only be added when clarification of the forward path is required. With these definitions, we are in a position to apply the topological equation.

In a closed network (a network consisting only of loops), there are no input nodes, and the set of independent linear equations describing the variables of the network are homogeneous. Hence the determinant of the matrix of coefficients is zero and the topological equation for a closed network is[13]

$$H(s) = 0 \quad \text{for all s.} \tag{5}$$

To apply Eq. 5, the network must be closed. If an equivalent w-function is desired between two nodes, then all nodes--excepting the two under consideration--which have no input branches or output branches can be omitted along with all branches incident to such nodes during the calculation of the particular w-function (the converse of superposition). This process is performed iteratively until there are no nodes other than the two under consideration which have no input branches or output branches. Now the network can be closed by adding a branch from the terminal node to the start node. This, by the definition of forward paths, will convert all forward paths between the two nodes to loops involving the two nodes. In this manner a network is closed by the addition of one branch. Examples of this process are given below.

Consider a network of arbitrary complexity and depict it by a black box as shown. Close the network by the addition of the required one branch,



By definition, the equivalent one-branch network from Q to T is $w_E(s)$. For this network there is only one loop, namely $w_E(s)w_A(s)$, and the topological equation yields

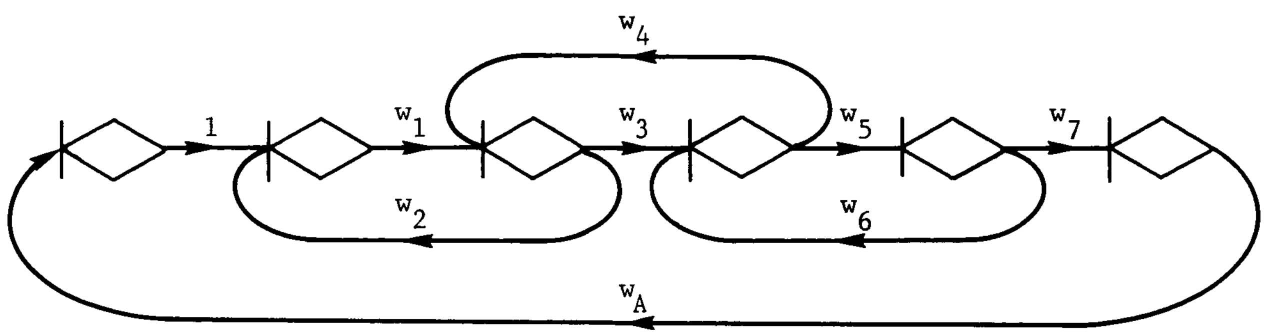
$$H(s) = 1 - w_E(s)w_A(s) = 0$$

and, hence,

$$w_A(s) = \frac{1}{w_E(s)} \quad (6)$$

This is a general result and the branch that is added to close a network is the inverse of the equivalent branch of the network.

Consider the more complex network discussed previously:



where w_A has been added. For this network there are now four loops of order 1, namely, $L_1(1) = w_1w_2$; $L_2(1) = w_3w_4$, $L_3(1) = w_5w_6$; and $L_4(1) = w_1w_3w_5w_Aw_7$; and one loop of order 2, $L_1(2) = w_1w_2w_5w_6$. From the topological equation, we have

$$H = 1 - w_1w_2 - w_3w_4 - w_5w_6 - w_1w_3w_5w_Aw_7 + w_1w_2w_5w_6 = 0 \ .$$

Solving for $w_E = 1/w_A$ yields

$$w_E = \frac{w_1w_3w_5w_7}{1 - w_1w_2 - w_3w_4 - w_5w_6 + w_1w_2w_5w_6} \ .$$

Since $w_A(s)$ or equivalently $w_E(s)$ is the quantity of interest, it is convenient to have an expression from which $w_E(s)$ can be computed directly. The equation for $H(s)$ is a linear form, i.e., the exponents of the $w_i(s)$ in each term will be either 0 or 1, and $H(s)$ can be written as a function of the terms not containing w_A and those terms which do contain w_A:

$$\left.\begin{aligned} H(s) &= H(s)\Big|_{w_A=0} + w_A \frac{\partial H(s)}{\partial w_A} = 0 \\ &\text{and} \\ w_E(s) &= \frac{1}{w_A(s)} = \frac{-\frac{\partial H(s)}{\partial w_A}}{H(s)\Big|_{w_{A=0}}} \,. \end{aligned}\right\} \qquad (7)$$

For the above example, we have

$$H(s)\Big|_{w_a=0} = 1 - w_1w_2 - w_3w_4 - w_5w_6 + w_1w_2w_5w_6$$

and

$$\frac{\partial H(s)}{\partial w_A} = - w_1w_3w_5w_7 \;,$$

and the equation for w_E results.

Careful examination of the above equation and example shows that $\frac{\partial H(s)}{\partial w_A}$ consists of each forward path between the nodes for which the equivalence is desired times one plus the loops disjoint from the forward path multiplied by (-1) raised to the order of the loop. $H(s)\Big|_{w_{A=0}}$ consists of all combinations of loops excluding the loop created by appending the branch $w_A(s)$. This equation for $w_E(s)$ is identical to Mason's rule or the loop rule for open graphs. Rewriting Eq. 7 in words we have

$$w_E(s) = \frac{\sum_i (\text{path } i)\,[1 + \sum_m (-1)^m (\text{loops of order } m \text{ not touching path } i)]}{[1 + \sum_m (-1)^m (\text{loops of order } m)]}$$

In Fig. 2, this equation is applied to the basic networks discussed previously where $w_j(s) = p_j e^{st_j}$, j = a, b, or c. Consider the self-loop network given in Fig. 2. The path from node 1 to node 2 is $w_a = p_a e^{st_a}$. The topological equation for the closed portion of the graph is $1 - w_b = 1 - p_b e^{st_b}$. Thus $w_E(s) = p_a e^{st_a}[1 - p_b e^{st_b}]^{-1}$. From the expression for $w_E(s)$, we have

$$p_E = w_E(0) = p_a(1 - p_b)^{-1}$$

and

$$M_E(s) = \frac{w_E(s)}{w_E(o)} = (1 - p_b)\, e^{st_a}\, [1 - p_b e^{st_b}]^{-1}.$$

Solving for μ_{1E} and μ_{2E} using Eq. 2, yields

$$\mu_{1E} = (1 - p_b)[t_a e^{st_a}(1 - p_b e^{st_b})^{-1} + e^{st_a}(1 - p_b e^{st_b})^{-2}\, p_b t_b e^{st_b}]_{s=0}$$

$$= [1 - p_b][t_a(1 - p_b)^{-1} + (1 - p_b)^{-2}\, p_b t_b]$$

$$= t_a + t_b\left(\frac{p_b}{1 - p_b}\right)$$

and

$$\mu_{2E} = (1 - p_b)[t_a^{\,2}\,(1 - p_b)^{-1} + t_a(1 - p_b)^{-2}\, p_b t_b + t_a(1 - p_b)^{-2}\, p_b t_b$$
$$+ 2(1 - p_b)^{-3}(p_b t_b)^2 + (1 - p_b)^{-2}\, p_b t_b^{\,2}]$$

$$\mu_{2E} = t_a^{\,2} + 2\, t_a t_b p_b(1 - p_b)^{-1} + 2(p_b t_b)^2(1 - p_b)^{-2} + p_b t_b^{\,2}(1 - p_b)^{-1}.$$

Using the above, the variance, VAR, of the time to traverse the path, given it is taken, is

$$\mathrm{VAR} = \mu_2 - (\mu_1)^2 = p_b t_b^{\,2}(1 - p_b)^{-1} + (p_b t_b)^2(1 - p_b)^{-2}$$

$$= p_b t_b^{\,2}(1 - p_b)^{-1}\, [1 + p_b(1 - p_b)^{-1}]$$

$$= p_b t_b^{\,2}(1 - p_b)^{-2}\ .$$

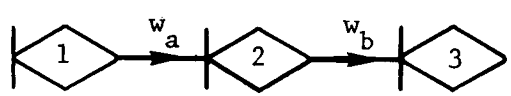
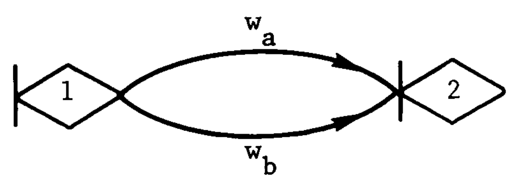
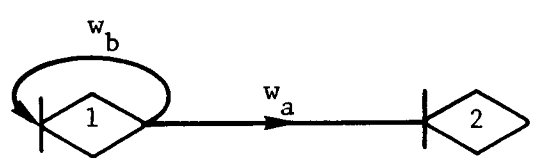
Network type	Graphical representation	Paths	Loops	Equivalent function w_E	Equivalent M.G.F. $M_E(s)$
Series		$w_a w_b$	--	$w_a w_b$	$e^{s(t_a + t_b)}$
Parallel		$w_a; w_b$	--	$w_a + w_b$	$\frac{1}{p_a + p_b} [p_a e^{st_a} + p_b e^{st_b}]$
Self-loop		w_a	w_b	$\frac{w_a}{1 - w_b}$	$(1 - p_b)e^{st_a}[1 - p_b e^{st_b}]^{-1}$

Fig. 2--Network reduction employing the topological equation

This same result is obtained by evaluating K_2 from Eq. 3. For comparison μ_2 can be obtained by enumeration of all paths multiplied by the square of the time to traverse the paths, viz.,

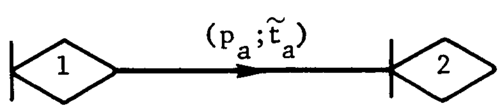
$$\mu_2 = \frac{1 - p_b}{p_a}\left[\sum_{j=0}^{\infty} p_a p_b{}^j (t_a + jt_b)^2\right],$$

where $\dfrac{1 - p_b}{p_a}$ is a normalizing factor.

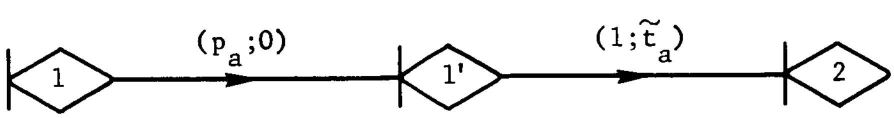
The discussion up to this point has been restricted to the situation in which all time intervals were considered as constants, and, hence, the variability or uncertainty in project duration was due entirely to the selection of branches in the realization of the network. In the next section, additional variability due to uncertainty in the time to traverse a branch that is realized will be included.

INCORPORATION OF RANDOM VARIABLES

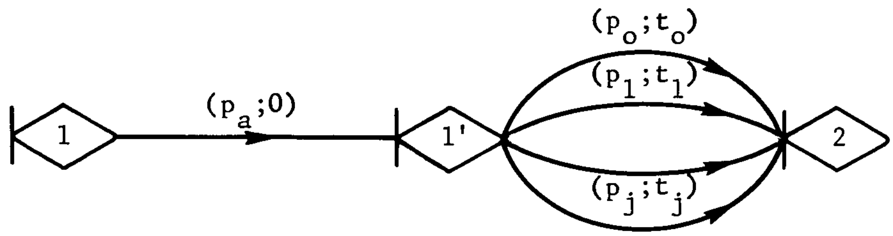
Consider the branch



where $\tilde{t}_a$ is a random variable. This branch can be represented as two branches in series, viz.,



Conceptually the second branch can be thought of as a set of parallel branches, each branch having a probability, p_j, of occurring with associated time, t_j, viz.,



where p_j and t_j are obtained from knowledge of the distribution of $\tilde{t}_a$. Now the t_j are constants and the analysis can proceed as discussed previously. For example, suppose $\tilde{t}_a$ is distributed according to a Poisson distribution with parameter λ. Then

$$p_j = P(t_j = j) = \frac{\lambda^j e^{-\lambda}}{j!} .$$

From Eq. 7 we obtain

$$w_E(s) = w_a(s) \sum_{j=0}^{\infty} w_j(s)$$

where $w_a(s) = p_a$ and $w_j(s) = p_j\ e^{sj}$.

Now

$$\sum_{j=0}^{\infty} w_j(s) = \sum_{j=0}^{\infty} p_j\ e^{st_j} = \sum_{j=0}^{\infty} \frac{\lambda^j e^{-\lambda}}{j!}\ e^{sj} = e^{\lambda(e^s-1)}$$

which is the M.G.F. for the Poisson distribution. This discussion leads to the important result that the M.G.F. can be substituted directly on the branch to describe the time parameter.* Thus random

---

*Since the parameter under consideration is time, which is an additive parameter, any transformation that causes the addition of two or more random variables to be a product of the transforms of the random variables is appropriate. Thus the Laplace transform and the Fourier transform (the characteristic function) would be acceptable. The M.G.F. was chosen because of its wide use and the fact that complex variable theory is not a prerequisite for its use. In Appendix B, the variables can be employed in the network without additional conceptual difficulties.

This result could have been the starting point for the analysis, since $M(s) = E\{e^{st}\}$, which is the reason for the e^{st} transform used previously. It is recognized that the M.G.F. for a constant, t, is e^{st} and, hence, the constant time situation is a special case of the above result. In the context used here $\tilde{t}$ is not limited to discrete values. If $\tilde{t}$ has a continuous distribution, then an integral would replace the summation operator and the M.G.F. would again result. Figure 3 presents the equivalent network for the simple combinations of the branches discussed previously. In Appendix A, Table 4, common moment generating functions are given.

Network type	Equivalent function, $w_E(s)$	Equivalent M.G.F. $M_E(s)$
(a) Series	$p_a p_b\, M_a(s)M_b(s)$	$M_a(s)M_b(s)$
(b) Parallel	$p_a M_a(s) + p_b M_b(s)$	$\left[\dfrac{1}{p_a + p_b}\right][p_a M_a(s) + p_b M_b(s)]$
(c) Self-loop	$p_a M_a(s)[1 - p_b M_b(s)]^{-1}$	$(1 - p_b)M_a(s)[1 - p_b M_b(s)]^{-1}$

Fig. 3--Reduction of networks with stochastic time intervals

Recall that the expressions given in Fig. 3 are for the basic networks. To show the power of GERT, suppose in Fig. 3 it is assumed that the M.G.F. for each branch is for the Poisson distribution. Then part (a) of Fig. 3 shows that two random variables in series each having

---

Laplace transform will be introduced to simplify inversion and complex convolution concepts. If the parameter is simply a count, then the generating function or equivalently the z-transform can be employed.[10]

a Poisson distribution result in an equivalent network whose time element also has a Poisson distribution. This is the same as saying the sum of two Poisson distributed random variables is Poisson distributed. An interpretation of part (b) of Fig. 3, assuming $p_a + p_b = 1$, is that the equivalent network is composed of samples taken from two Poisson distributions where the values are drawn in the ratio of p_a to p_b. Thus the equivalent is a mixture of two Poisson distributions which is not Poisson.[13,14,15] Note that in part (b) of Fig. 3 no assumptions regarding the distributions of $\tilde{t}_a$ and $\tilde{t}_b$ were made and mixtures of any distribution can be studied. Although the loop network in part (c) of Fig. 3 appears as a simple network, it represents a complex stochastic process, namely a sum of a random number of independently distributed random variables. This interpretation results when it is considered that the number of times the self-loop is realized is a random variable, as is the time interval realized on each transfer about the loop. In a later section a more detailed analysis of this network will be presented.

In the preceding paragraphs it has been assumed that the times on the branches are independent random variables. Throughout this Memorandum this assumption will be made. If the times are not independent, then the M.G.F. will have to be conditioned not only on the branch being realized, but also on the time for which the branch is dependent.

In the next three sections the procedures discussed in this section will be illustrated and applied.

IV. EVALUATION OF COMPLEX NETWORKS

In this section GERT will be used to evaluate: (1) a network with multiple loops; and (2) a network with multiple input and output nodes. Specific applications are discussed in the following two sections; thus the emphasis here is entirely on the mechanics of the evaluation portion of GERT with a minimum of descriptive material.

Recall that from the $w_E(s)$ function the probability of realizing the output node is obtained from $p_E = w_E(o)$ and that the n^{th} moment of the time to traverse the network is given by $\mu_{nE} = \left.\frac{\partial^n}{\partial s^n}\left(\frac{w_E(s)}{w_E(o)}\right)\right|_{s=o}$.

In this section only the $w_E(s)$ function will be derived. The use of the information contained in the $w_E(s)$ function is dependent on the application and the system of which the stochastic network is a model.

EXAMPLE 1. MULTIPLE FEEDBACK LOOPS

Consider the network given in Fig. 4:

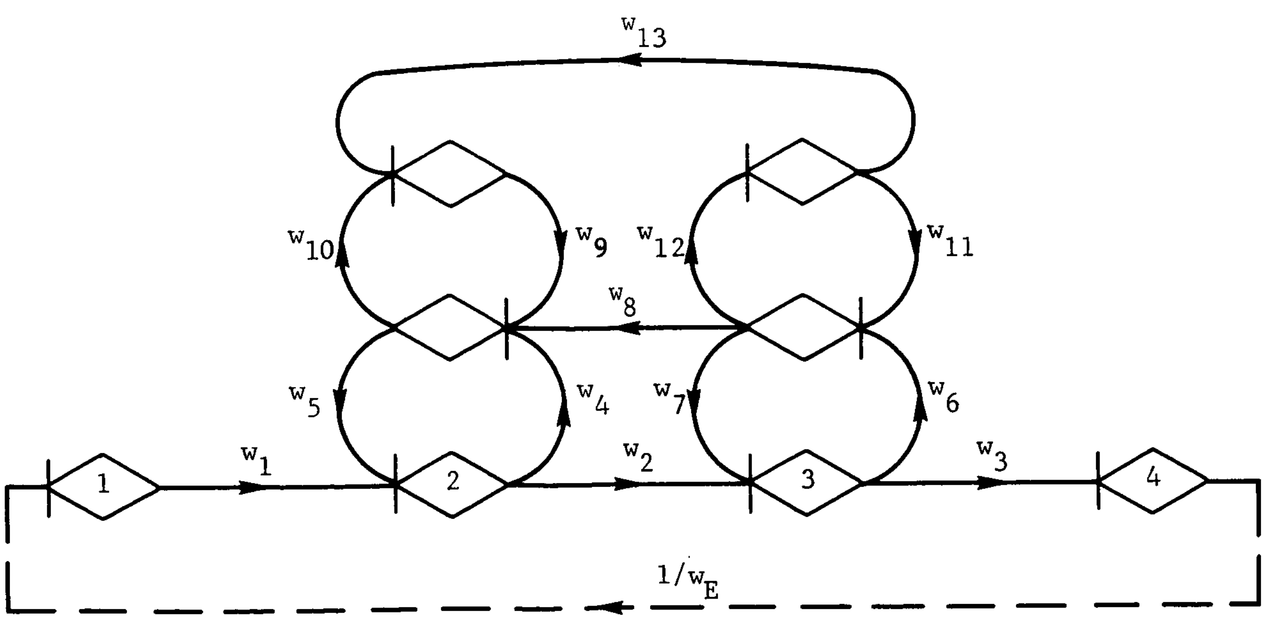


Fig. 4--A complex network

In Table 1 a list of the loops of order 1, 2 (nontouching pairs), and 3 (nontouching triples) is presented:

Table 1

LISTING OF LOOPS FOR COMPLEX NETWORK OF FIGURE 4

Loop	Elements of Loop	Nontouching Associated Loops	Nontouching Association of Three Loops
L_1	$w_1 w_2 w_3 \frac{1}{w_E}$	L_5, L_6	$L_5 L_6$
L_2	$w_4 w_5$	L_3, L_6	-
L_3	$w_6 w_7$	L_2, L_5	-
L_4	$w_2 w_6 w_8 w_5$	-	-
L_5	$w_9 w_{10}$	L_3, L_6, L_1	$L_1 L_6$
L_6	$w_{11} w_{12}$	L_2, L_5, L_1	$L_1 L_5$
L_7	$w_2 w_6 w_{12} w_{13} w_9 w_5$	-	-

From Table 1 and the topological equation, we have

$$H(s) = 1 - \sum_{i=1}^{7} L_i + L_1L_5 + L_1L_6 + L_2L_3 + L_2L_6 + L_3L_5 + L_5L_6 - L_1L_5L_6 = 0.$$

At this point the problem has been reduced to algebraic manipulations identical to the standard signal flowgraph manipulations. The resulting equation for w_E (suppressing the s for convenience) is

$$w_E = w_1 w_2 w_3 \left[\frac{1 - w_9w_{10} - w_{11}w_{12} + w_9w_{10}w_{11}w_{12}}{H(s)\big|_{1/w_E = 0}} \right]$$

where $H(s)\big|_{1/w_E = 0} =$

$$1 - (w_4w_5 + w_6w_7 + w_2w_6w_8w_5 + w_9w_{10} + w_{11}w_{12} + w_2w_6w_{12}w_{13}w_9w_5)$$

$$+ w_4w_5(w_6w_7 + w_{11}w_{12}) + w_6w_7w_9w_{10} + w_9w_{10}w_{11}w_{12}.$$

Mason's Loop rule could have been employed and w_E obtained directly from the graph illustrated in Fig. 4. A digital computer program has been written that computes the probability that an output node is realized and the first two moments of the time to realize the output node, given that it is realized. The program is discussed in Appendix A. This program makes the analysis of larger problems purely mechanical.

EXAMPLE 2. MULTIPLE INPUT AND OUTPUT NODES

The next example involves multiple input and output nodes, as illustrated in Fig. 5:

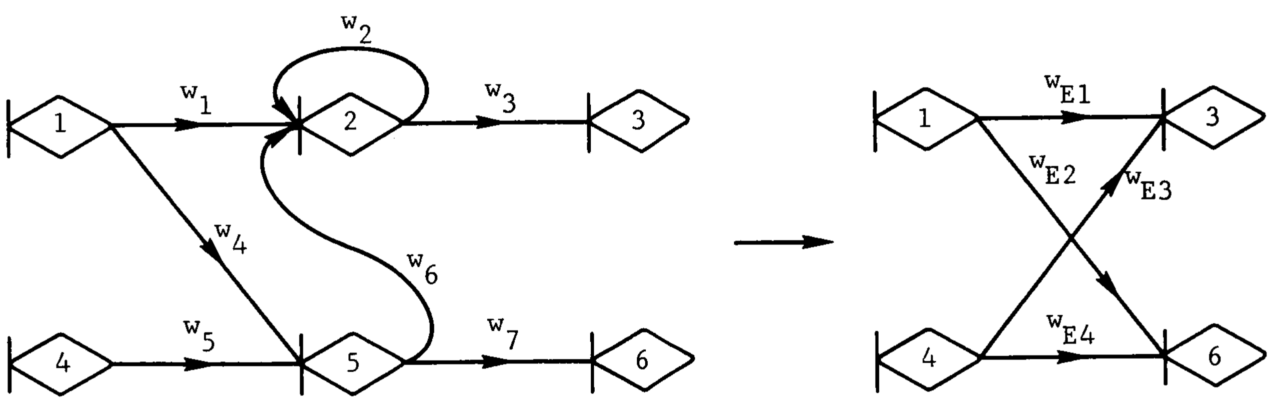


Fig. 5--A network with multiple input and output nodes

From Fig. 5, the following equivalent branch equations can be obtained by using Eq. 7:

$$w_{E1} = \frac{w_3(w_1 + w_4w_6)}{1 - w_2}$$

$$w_{E2} = w_4w_7$$

$$w_{E3} = \frac{w_3w_5w_6}{1 - w_2}$$

and

$$w_{E4} = w_5w_7 .$$

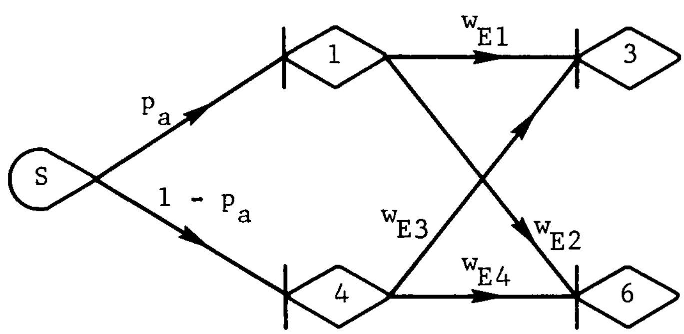
The probabilities of the branch being part of a realization and the moments of the times associated with an equivalent branch, if it is part of the realization, are computed from w_{Ej} as previously discussed.

Suppose it is given that p_a proportion of the time node 1 is the starting node and $(1 - p_a)$ proportion of the time node 4 is the starting node. Given this information we can write directly that

$$w_{s3} = p_a w_{E1} + (1 - p_a) w_{E3}$$

and
$$w_{s6} = p_a w_{E2} + (1 - p_a) w_{E4}.$$

This can be seen from the following network:



The relationship between this network (including nodes 2 and 5) and Markov chains is seen in the transition probability matrix given below, where a blank indicates a zero entry:

	S	1	2	3	4	5	6
S		p_a			$1 - p_a$		
1			p_1			p_4	
2			p_2	p_3			
3				1			
4						p_5	
5			p_6				p_7
6							1

A network is seen to represent a sparse transition probability matrix. The stochastic network also includes the concept of transition time. For the above matrix it was assumed that if either node 3 (state 3) or node 6 (state 6) was realized then the network would be realized. Thus nodes 3 and 6 represent absorbing states and once reached, the process never leaves these nodes. This concept corresponds to a self-loop about nodes 3 and 6 or a "1" in diagonal of the above matrix in rows 3 and 6.

A stochastic network corresponds closely to processes of the semi-Markov variety.(16) The main theorems of semi-Markov processes pertain to processes whose underlying Markov chain is ergodic. In network terminology an ergodic chain is one in which every node can be reached from every other node in a finite number of branch transitions. Quantities of interest in semi-Markov process theory are the steady-state probability of being in a particular state, and the steady-state percentage of time spent in a particular state where a state includes all activities represented by the branches leaving a node. From these quantities other pertinent information can be obtained, such as mean recurrence time of a state. For stochastic networks, the states are normally transient and the basic theorems of semi-Markov processes are not of interest. Whitehouse[17] has developed techniques by which GERT can be used to analyze stochastic networks that are ergodic in the Markov chain sense. He has applied these techniques to analyze inventory and queueing problems.

V. APPLICATION OF GERT TO PROBABILITY PROBLEMS*

In this section GERT will be applied to:

1. The development of moment generating functions for several probability laws.

2. The solution of complex probability problems.

3. The analysis of stochastic processes.

The application of GERT not only solves these problems but transforms a relatively inductive process into a deductive and algebraic procedure. The benefits from a teaching standpoint are significant.

DEVELOPMENT OF FAMILIAR MOMENT GENERATING FUNCTIONS

Example 3. The Geometric Probability Law

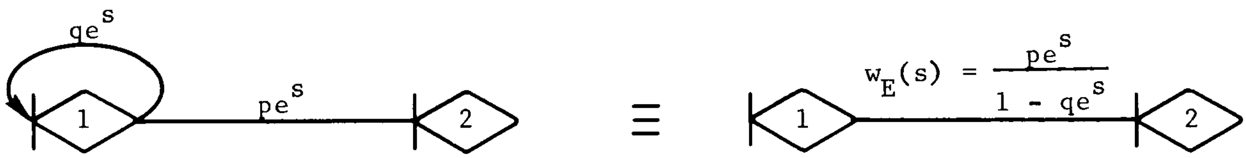
Consider the problem of determining the number of trials required to obtain the first success in a sequence of Bernoulli trials in which the probability of success at each trial is p. The number of trials is a random variable which obeys the Geometric Probability law.

To represent this problem in network form, define two nodes: let node 2 represent the event "first success" and node 1 the event "no successes." The problem can be put in network form by realizing that the process starts at trial number zero with no successes (node 1) and can on one trial stay in (or return to) the state of no successes (node 1) with probability q, or have a first success (go to node 2) with probability p. The time it takes for either move is one trial or time unit.

---

*This section contains theoretical applications of GERT and could be bypassed by readers interested solely in practical applications.

Thus, the stochastic network for this process is:



The M.G.F. of the number of trials required to obtain the first success is

$$M_E(s) = \frac{w_E(s)}{w_E(o)} = \frac{pe^s}{1 - qe^s}$$

where $w_E(0) = 1$ since $p + q = 1$. $M_E(s)$ is recognized as the M.C.F. of the geometric distribution. In this example, the second parameter of the ordered pair was a trial, not a time interval, i.e., each success or failure corresponds to one trial. GERT permits a variable time interval for a success or failure. Thus if the time for a trial is Poisson distributed with parameter λ_a for a success and λ_b for a failure, then

$$w_E(s) = \frac{pe^{\lambda_a(e^s-1)}}{1 - qe^{\lambda_b(e^s-1)}} .$$

From this expression it is seen that $p_E = 1$, i.e., node 2 is always realized, and $M_E(s) = w_E(s)$. The mean and variance of the time to obtain the first success are given by

$$K_1 = \frac{\partial}{\partial s} [\ell n\ M_E(s)]_{s=o} = \lambda_a + \frac{q}{p}\lambda_b$$

and

$$K_2 = \frac{\partial^2}{\partial s^2} [\ell n\ M_E(s)]_{s=o} = \lambda_a + \frac{q}{p}\lambda_b(\lambda_b + 1) + (\frac{q}{p}\lambda_b)^2$$

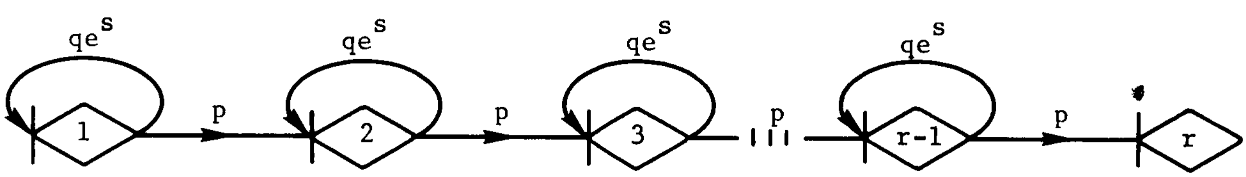
For the case where the second parameter is a trial, Huggins[10] analyzed the problem by employing the probability generating function. This is equivalent to substituting $x=e^s$ in the above graph which yields

$$w_E(x) = \frac{px}{1 - qx} \quad .$$

The use of probability generating functions is described in detail by Feller.[18]

Example 4. The Negative Binomial Probability Law

A more complex problem is the determination of the number of failures encountered before the r^{th} success in a sequence of independent Bernoulli trials. The stochastic network is



Note that since only failures are counted, the number of trials for a success is set equal to zero. In this problem there are r nontouching self-loops and the topological equation is:

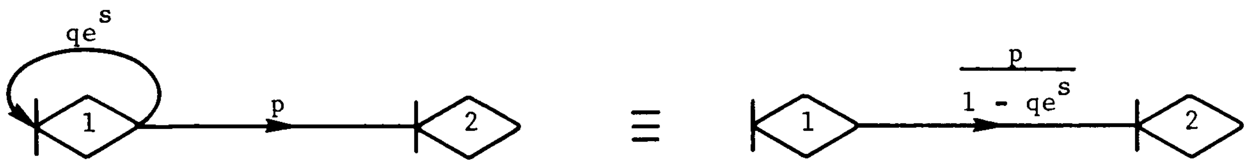
$$H(s) = 1 - \frac{(p)^r}{w_E(s)} + \sum_{j=1}^{r} \binom{r}{j} (-1)^j (qe^s)^j = 0.$$

Solving for $w_E(s)$ by employing the binomial expansion yields

$$w_E(s) = \left(\frac{p}{1 - qe^s}\right)^r = M_E(s) \qquad \text{since} \quad w_E(o) = 1.$$

$M_E(s)$ is recognized as the M.G.F. of the negative binomial as expected.

A more direct procedure for solving this stochastic network is to reduce the network in segments. Consider the basic element as



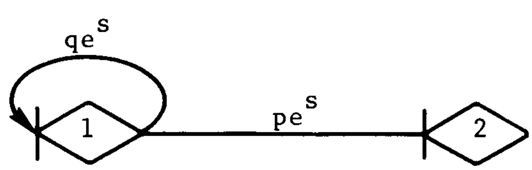
From the above it is seen that there are r of these equivalent branches in series and

$$w_E(s) = M_E(s) = \left(\frac{p}{1-qe^s}\right)^r \quad .$$

This example demonstrates the importance in some cases of employing the reduction procedure first for part of the network and then for the entire network.

<u>Example 5. A Modified Negative Binomial Probability Law</u>

As an extension of Example 4, consider the distribution of the number of trials required before the r^{th} success. In this case the network is a series of r subnetworks of the form

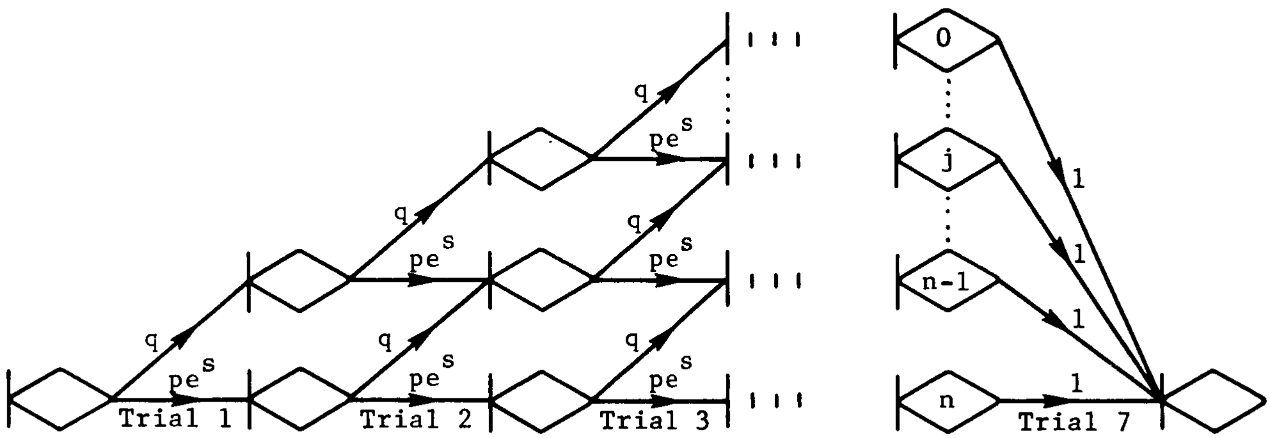


and

$$w_E(s) = M_E(s) = \left(\frac{pe^s}{1-qe^s}\right)^r \quad .$$

Example 6. The Binomial Probability Law

As a last example in this subsection, consider the distribution of the number of successes in n independent Bernoulli trials. The network is



and it contains no feedback loops. After trial n there have been 0,1,2,..., n successes represented by the (n+1) nodes preceding the terminal node. Since these outcomes are mutually exclusive, the (n+1) nodes can be connected to a single output node. This permits the distribution of the number of successes in n trials to be obtained. The topological equation for this network is

$$H(s) = 1 - \frac{1}{w_E(s)} \sum_{j=0}^{n} \binom{n}{j} (pe^s)^j q^{n-j} = 0$$

and

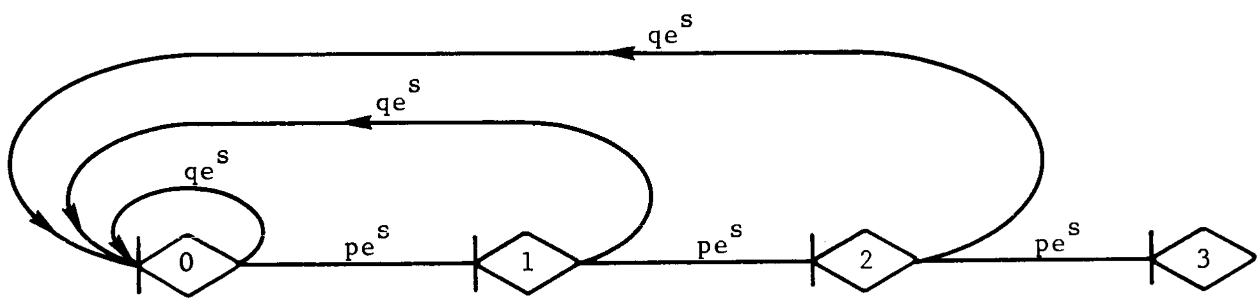
$$w_E(s) = M_E(s) = (pe^s + q)^n \quad \text{as expected.}$$

SOLUTION OF PROBABILITY PROBLEMS

The application of GERT to selected probabilistic problems will be discussed below, including the drawing of the network and the derivation of the equivalent network equations.

Example 7. Dice Throwing

Consider the problem of determining the number of throws of a pair of dice required to obtain three consecutive sevens if the probability of obtaining a seven is p and the probability of not obtaining a seven is q. The network for this problem is



For this network

$$w_E(s) = \frac{(pe^s)^3}{1-qe^s-pq(e^s)^2 - p^2q(e^s)^3}$$

$$= \frac{(pe^s)^3 (1-pe^s)}{1-e^s+(pe^s)^3 (1-pe^s)} \quad .$$

Since $w_E(0) = 1$, $M_E(s) = w_E(s)$ and the M.G.F. for this specific problem is obtained. Extension to the general case of n consecutive values of seven (or any other possible number) is straightforward with the result that

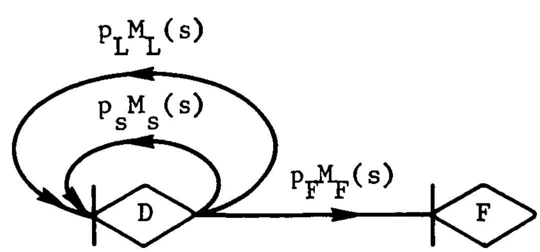
$$w_E(s) = M_E(s) = \frac{(pe^s)^n (1-pe^s)}{1-e^s + (pe^s)^n (1-pe^s)} \quad .$$

For example, the M.G.F. of the number of throws to obtain 10 sevens where the probability of obtaining a seven is 1/6 is

$$M_E(s) = \frac{(1/6\ e^s)^{10}(1 - 1/6\ e^s)}{1 - e^s + (1/6\ e^s)^{10}(1 - 1/6\ e^s)}$$

Example 8. The Thief of Bagdad.

The following problem has been abstracted from Parzen[19]. The thief of Bagdad has been placed in a dungeon (node D) with three doors. One door leads to freedom (node F), one door leads to a long tunnel, and a third door leads to a short tunnel. The tunnels return the thief to the dungeon (node D). If the thief returns to the dungeon he attempts to gain his freedom again, but his past experience(s) do not help him in selecting the door that leads to freedom, i.e., the probabilities associated with the thief's selection of doors remain constant. Let p_F, p_S, and p_L denote the thief's probabilities of selecting the doors to freedom, the short tunnel, and the long tunnel, respectively. The network for this problem is



and

$$w_E(s) = M_E(s) = \frac{p_F M_F(s)}{1 - p_S M_S(s) - p_L M_L(s)} .$$

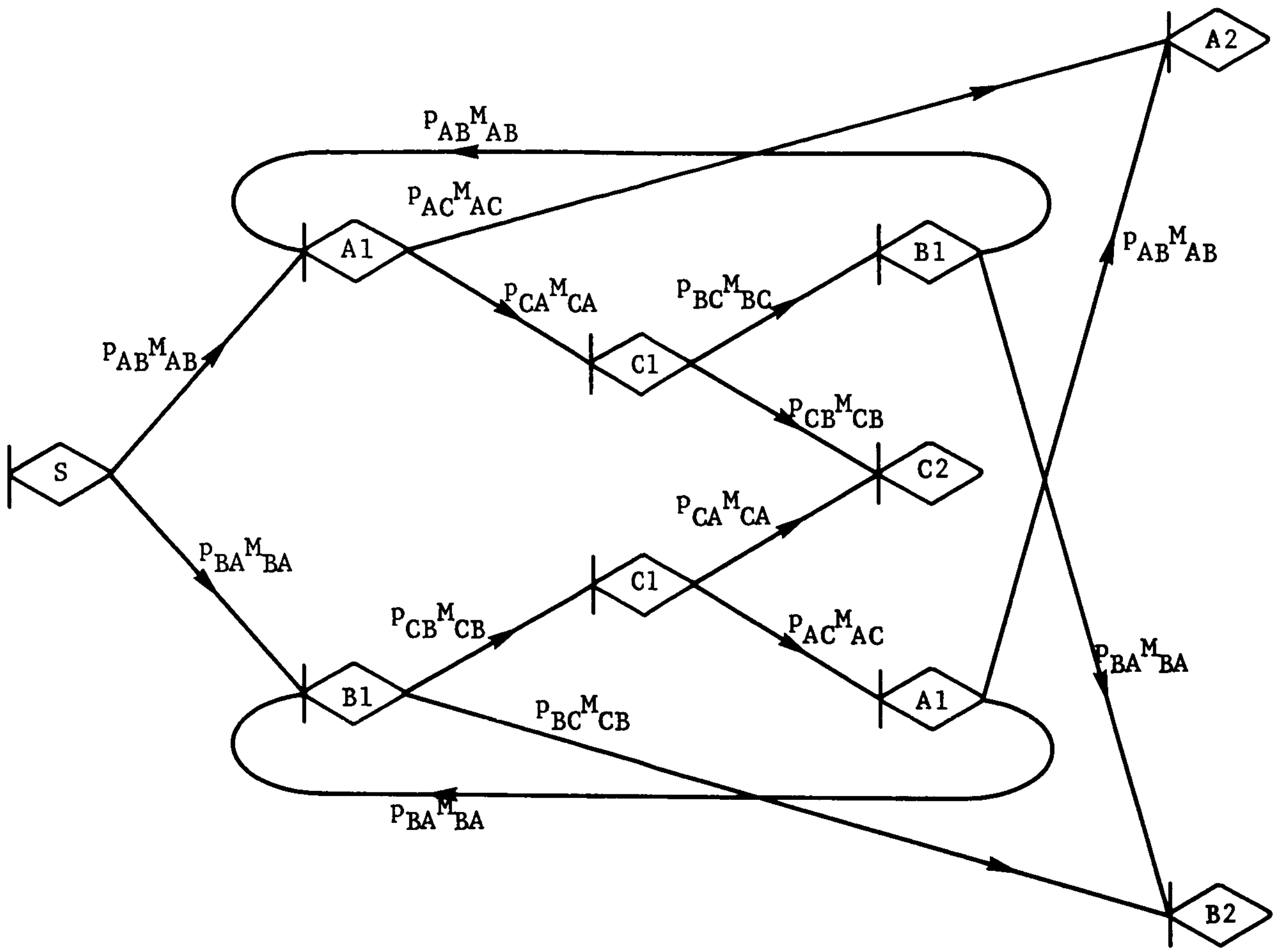
From this equation for $M_E(s)$, the moments with regard to the time it takes for the thief to reach freedom are completely characterized. For the example given in Parzen, $p_F = p_s = p_L = 1/3$ and $M_F(s) = e^o = 1$, $M_L(s) = e^{3s}$ and $M_s(s) = e^s$. Thus the expected time for the thief to obtain his freedom is

$$\mu_1 = t_F + \frac{1}{p_F} [p_L t_L + p_s t_s] = 4 \text{ time units.}$$

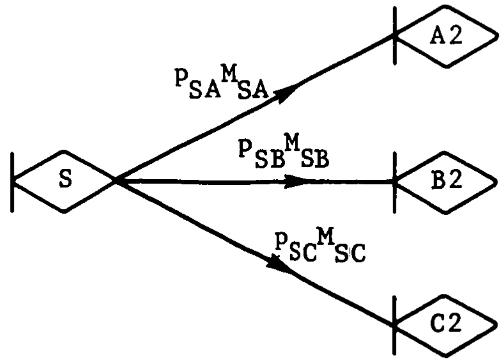
It is interesting to note that the introduction of random variables for the times associated with each activity would not alter the formulation part of this problem. Only the algebraic manipulations are increased.

Example 9. A Three Player Game

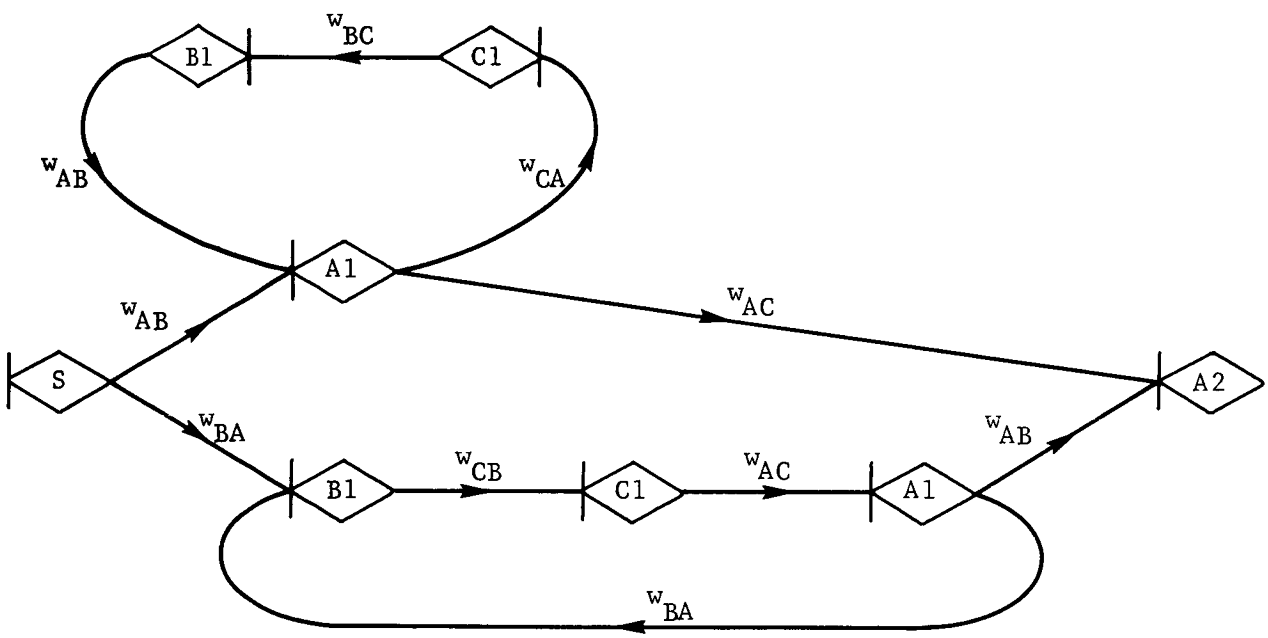
Let us examine another problem taken from Parzen.[20] Three players (denoted by A, B, and C) take turns playing a game according to the following rules: at the start A and B play while C sits out; the winner of the match between A and B then plays C; the winner of the second match then plays the loser of the preceding match until a player wins twice in succession, in which case he is declared the winner of the game. The network for this game is given below:



where p_{IJ} denotes the probability that I defeats J; I, J = A,B,C and M_{IJ} denotes the M.G.F. of the time required for I to defeat J. The resulting equivalent network for this game is



The above three branches of the resulting equivalent network can be examined separately. Consider the branch from S to A2. Redrawing the network we have



and

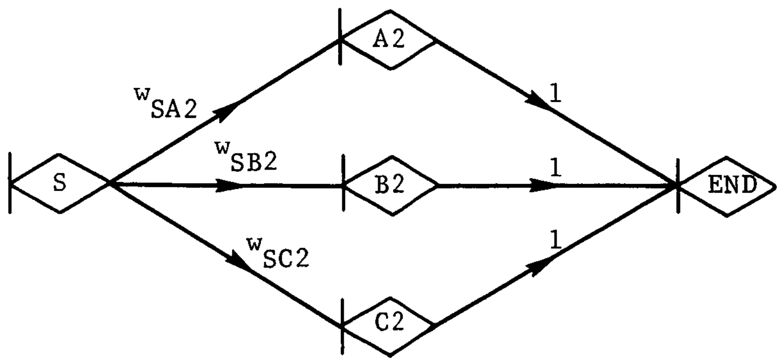
$$w_{SA2} = \frac{w_{AB}w_{AC}(1-L_{BA}) + w_{BA}w_{CB}w_{AC}w_{AB}(1-L_{AA})}{(1-L_{BA})\ (1-L_{AA})}$$

where $L_{AA} = w_{CA}w_{BC}w_{AB}$ and $L_{BA} = w_{CB}w_{AC}w_{BA}$, where the topological equation for an open graph of two disjoint loops is $[1-L_1(1)][1-L_2(1)]$. The expression for w_{SB2} will be identical to the above with B and A interchanged. For the branch from S to C2, we have

$$w_{SC2} = \frac{w_{AB}w_{CA}w_{CB}(1-L_{BA}) + w_{BA}w_{CB}w_{CA}(1-L_{AA})}{(1-L_{BA})\ (1-L_{AA})}$$

The procedures for obtaining the M.G.F. can now be applied. In Appendix A this problem is analyzed using a digital computer program.

If it is desired to obtain the M.G.F. of the time to the end of the game, then we add three branches to the resulting network and obtain



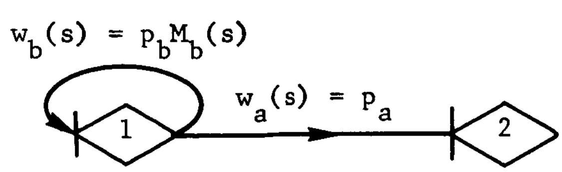
Then

$$w_{SEND}(s) = w_{SA2} + w_{SB2} + w_{SC2} = M_{SEND}(s)$$

since the probability that the game ends is one ($w_{SEND}(0) = 1$).

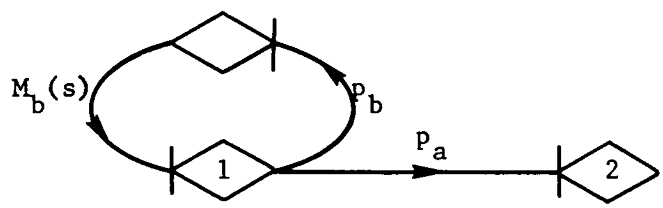
A RANDOM NUMBER OF RANDOM VARIABLES

As discussed previously, the time to traverse a simple network consisting of a single self-loop and one forward branch is a specific example of a random variable, which is a sum of identically distributed random variables where the number of terms in the sum is itself a random variable. Consider the network



with $p_a = 1 - p_b$.

Breaking the self-loop into two parts for discussion purposes we have



From this illustration it is seen that the self-loop is taken once with probability p_b, twice with probability p_b^2, and n times with probability p_b^n. Each time the self-loop is traversed a time is drawn from the distribution of $\tilde{t}_b$. Since the time to traverse the "a" branch is zero, the time to go from node 1 to node 2 is the sum of the times resulting from traversals of the self-loop. With this interpretation it is seen that the self-loop does portray a specific type of random variable, which is a sum of a random number of identically distributed random variables. The equation for the variance of a random sum of random variables will be verified by analysis of the above network.

From the network it is seen that

$$w_E(s) = M_E(s) = p_a\,[1 - p_b M_b(s)]^{-1} \quad \text{since } w_E(s) = 1.$$

Taking natural logarithms and finding the first two moments about the mean results in the following:

$$\ell n\, M_E(s) = \ell n\, p_a - \ell n[1 - p_b M_b(s)]$$

$$\frac{d}{ds}\,[\ell n\, M_E(s)] = p_b M_b'(s)[1 - p_b M_b(s)]^{-1}$$

where

$$M_b'(s) = \frac{\partial M_b(s)}{\partial s} \quad .$$

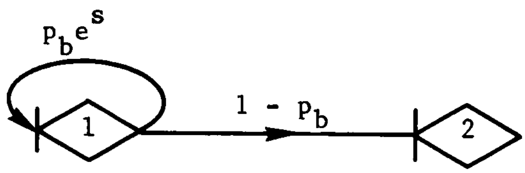
Setting $s = 0$ yields the first moment, K_1

$$K_1 = p_b M_b'(o)[1 - p_b M_b(o)]^{-1}$$

$$= \frac{p_b}{1 - p_b} M_b'(o)$$

$$= E\{N\}E\{t_b\}$$

where N is the number of times the self-loop is traversed. To compute $E\{N\}$ we apply GERT to the network



which yields $M_N(s) = (1 - p_b)(1 - p_b e^s)^{-1}$ and $E\{N\} = \dfrac{p_b}{1 - p_b}$ and $VAR\ \{N\} = E^2\{N\} + E\{N\}$. The second moment about the mean, K_2, is obtained as

$$K_2 = \frac{d^2}{ds^2}[\ell n\ M_e(s)]_{s=o} = \Big\{[p_b M_b'(s)]^2[1 - p_b M_b(s)]^{-2}$$

$$+ p_b M_b''(s)[1 - p_b M_b(s)]^{-1}\Big\}_{s=o} = E^2\{N\}E^2\{t_b\} + E\{N\}E\{t_b{}^2\}.$$

To show that this is equivalent to the standard form,(19) we write

$$K_2 = E^2\{N\}E^2\{t_b\} + E\{N\}[Var\{t_b\} + E^2\{t_b\}]$$

$$= E^2\{t_b\}[E^2\{N\} + E\{N\}] + E\{N\}Var\{t_b\}$$

$$= E^2\{t_b\}Var\{N\} + E\{N\}Var\{t_b\} \quad ,$$

and the desired expression for K_2 is obtained. Higher cumulants can be calculated directly by differentiation of the $\ell n\, M_e(s)$.

This problem demonstrates the power of GERT in that the M.G.F. of a system which consists of many self-loops and combinations of self-loops can be obtained directly using the topological equation.

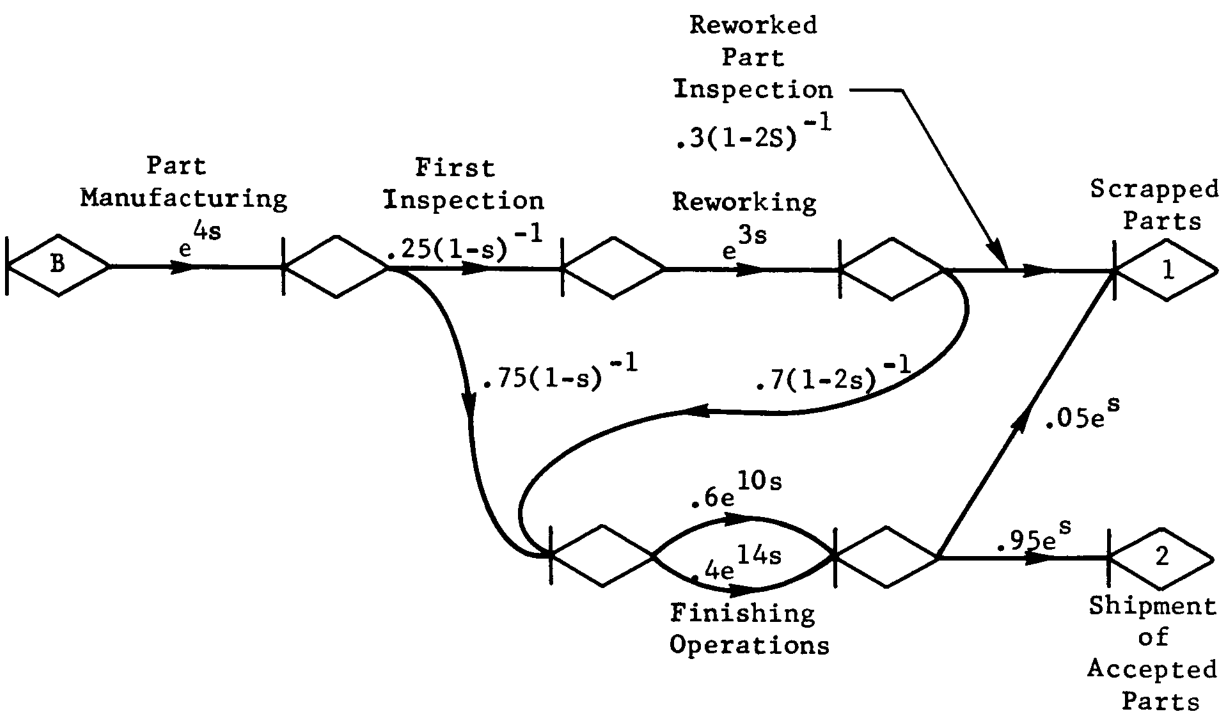
VI. APPLICATION OF GERT TO ENGINEERING PROBLEMS

In this section the emphasis will be on the description of the problem and the development of the stochastic network portrayal of the problem. The discussion of the procedure for obtaining equivalent networks will be held to a minimum.

EXAMPLE 10. A MODEL OF A MANUFACTURING PROCESS

GERT can be employed to model and analyze manufacturing processes if it is assumed that steady-state conditions exist and that the M.G.F. of the times spent at an operation center can be obtained. The level of aggregation of the times at an operation center depends on the availability of data. Two levels of aggregation would be: (1) a single M.G.F. defining the time from receipt at the center to the time it was removed from the center; and (2) two M.G.F. representing a waiting time distribution and a service time distribution. Either level of aggregation is amenable to GERT, as will be discussed in this example.

On a production line a part is manufactured at the beginning of the line. The manufacturing operation is assumed to take four hours. Before the finishing operations are started on the part, it is inspected--with 25 per cent of the parts failing the inspection and requiring rework. The inspection time (including waiting for inspection) is assumed to be distributed according to the negative exponential distribution, with a mean of one hour. Reworking takes three hours, and 30 per cent of the parts reworked fail the next inspection. This inspection of reworked items is also distributed according to the negative exponential, with a mean of one-half hour. Parts which fail this inspection are scrapped. If the part passes either of the above inspections, it is sent to the final finishing operation, which takes 10 hours 60 per cent of the time and 14 hours 40 per cent of the time. A final inspection, which takes one hour, rejects 5 per cent of the parts, which are then scrapped. The stochastic network for this production line is:



From the network we have

$$w_{B1}(s) = e^{4s}(.25)(1 - s)^{-1}(e^{3s})(.3)(1 - 2s)^{-1}$$

$$+ e^{4s}(.25)(1 - s)^{-1}e^{3s}(.7)(1 - 2s)^{-1}[.6e^{10s} + .4e^{14s}](.05)e^{s}$$

$$+ e^{4s}(.75)(1 - s)^{-1}[.6e^{10s} + .4e^{14s}](.05)e^{s}$$

and

$$w_{B2}(s) = e^{4s}[(.25)(1 - s)^{-1}e^{3s}(.7)(1 - 2s)^{-1}$$

$$+ (.75)(1 - s)^{-1}][.6e^{10s} + .4e^{14s}](.95)e^{s} .$$

Setting s = o we have

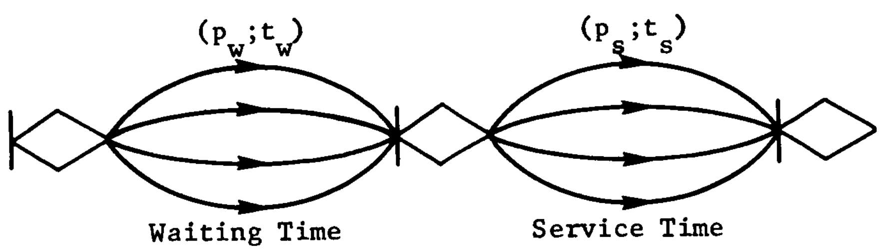
$$w_{B2}(0) = [(.25)(.7) + (.75)](.95) = 0.87875$$

and

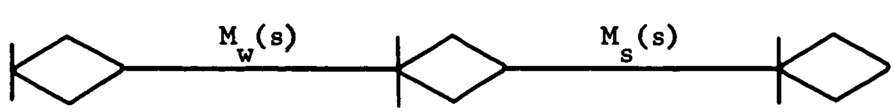
$$w_{B1}(0) = 0.12125.$$

From previous developments we know that $M_{B2}(s) = \frac{w_{B2}(s)}{w_{B2}(o)}$ and $M_{B1}(s) = \frac{w_{B1}(s)}{w_{B1}(o)}$. These two M.G.F. characterize the distribution of the time required for a part to flow through the production line. Although transit times were not included in this example, they are merely a simple addition.

It is interesting to note that in the above example waiting time and service time were aggregated into one distribution. A more realistic pictorial representation of a single channel queueing system is



where a part (unit, or customer) arrives and will have a waiting time obtained from a waiting time distribution and then a service time obtained from the service time distribution. In stochastic network form this reduces to

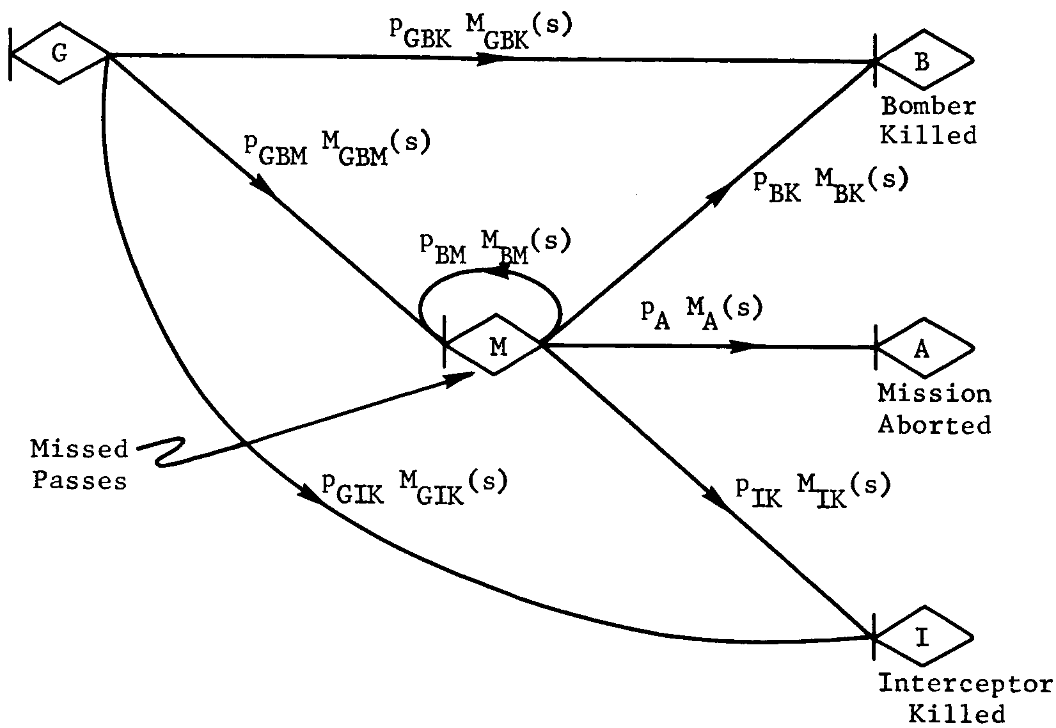


where it is assumed that both paths must be taken (one of the waiting times is zero). The stochastic network shown above can be cascaded to form tandem queueing systems. The problem involved in solving this cascaded network will be input availability, since the waiting time distribution for the second, third, and higher service stations is difficult to derive. If the input data is available then it is a simple application of GERT to obtain results (such as the distribution of the time through the system) for networks of queueing systems.

EXAMPLE 11. WAR GAMING

In this example, a simple air duel model will be structured in stochastic network form and GERT applied for evaluation purposes.

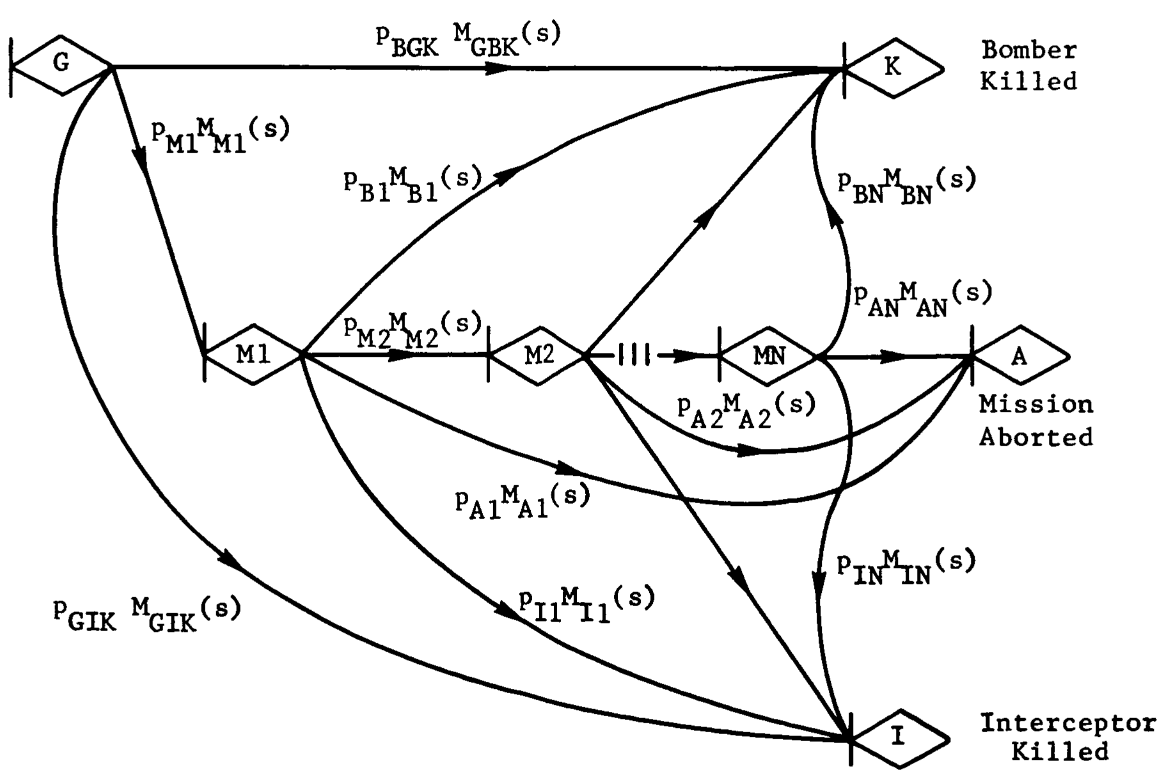
An interceptor is alerted and is assigned a specific bomber as its target. The time for the interceptor to climb to altitude, approach the bomber, and make a pass is a random variable with M.G.F., $M_{GBK}(s)$, if the interceptor shoots down (kills) the bomber. If the interceptor misses, then it will be assumed that the time taken is from the distribution whose M.G.F. is $M_{GBM}(s)$. There is a third possibility, viz., the bomber will shoot down the interceptor on the first pass. The M.G.F. for this case is symbolized by $M_{GIK}(s)$. If the interceptor misses, then there are successive passes made at the bomber; however, after each pass there is a probability that the interceptor's mission will have to be aborted. First an infinite number of passes will be considered, then a restriction on the number of passes will be imposed. The stochastic network is



$$\text{and } w_{GB}(s) = \frac{p_{GBK}\ M_{GBK}(s)[1 - p_{BM}\ M_{BM}(s)] + p_{GBM}\ M_{GBM}(s) p_{BK}\ M_{BK}(s)}{[1 - p_{BM}\ M_{BM}(s)]} .$$

The expressions for $w_{GI}(s)$ and $w_{GA}(s)$ are computed in a similar manner.

If the number of passes is restricted and the probabilities and distributions of times change for each pass, then the network would be:



For this network we have

$$w_{GK}(s) = p_{GBK}\ M_{GBK}(s) + \sum_{j=1}^{N} \left\{ \left[\prod_{i=1}^{j} p_{Mi}M_{Mi}(s) \right] p_{Bj}M_{Bj}(s) \right\} ,$$

$$w_{GI}(s) = p_{GIK}\ M_{GIK}(s) + \sum_{j=1}^{N} \left\{ \left[\prod_{i=1}^{j} p_{Mi}M_{Mi}(s) \right] p_{Ij}M_{Ij}(s) \right\}$$

and

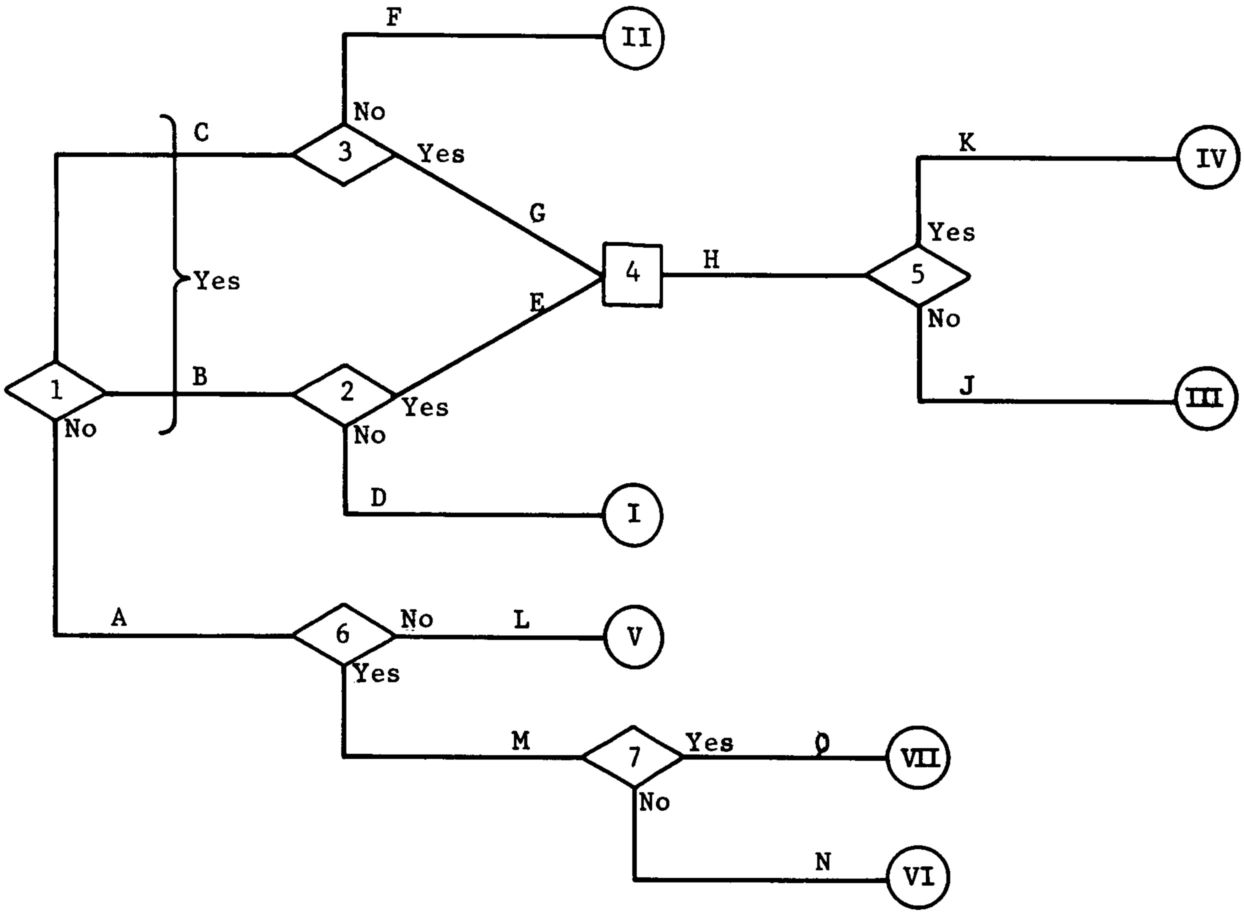
$$w_{GA}(s) = \sum_{j=1}^{N} \left\{ \left[\prod_{i=1}^{j} p_{Mi}M_{Mi}(s) \right] p_{Aj}M_{Aj}(s) \right\} .$$

More complex air duel situations can be modeled along the lines prèsented in this example.

EXAMPLE 12. ANALYSIS OF RESEARCH AND DEVELOPMENT EXPENDITURES

Eisner[5] suggested the introduction of probabilistic elements on PERT networks in order to study R&D problems. In a recent article, Graham,[21] using the concepts presented by Eisner, derives the network as shown in Fig. 6. For each branch of the network, Graham gives the probability that the branch is realized, given that the preceding node is realized, and the time and cost (assumed to be constants by Graham) associated with the activity represented by the branch if the activity is performed. These values are inserted on the GERT network in Fig. 7 for this problem by an ordered triple of: probability; time (weeks); and cost in $1000 units; viz., (p, t, c). Time in this example is not a duration but the amount of effort required to perform the activities measured in weeks.

Several changes were made in the construction of the GERT network. First, the AC and DC control investigations (activities B and C) are



Events
1 Feasibility study indicates electrical control of high temperature system is/is not feasible.
2 AC control found suitable/unsuitable.
3 DC control found suitable/unsuitable.
4 Optimum integration of AC/DC circuits achieved.
5 Unit found to be within/outside potential market price.
6 Pneumatic control found to be feasible/unfeasible.
7 Unit found to be within/outside potential market price.

Activities
A Pneumatic feasibility study.
B AC control investigation.
C DC control investigation.
D Report writing.
E Investigation of optimum AC/DC integration.
F Report writing.

Activities (continued)
G Investigation of optimum AC/DC integration.
H Economic analysis of system.
J Report writing.
K Report writing.
L Report writing.
M Economic analysis of system.
N Report writing.
O Report writing.

Outcomes
I Project dropped.
II Project dropped.
III Project dropped.
IV Product put into production and marketed.
V Project dropped.
VI Project dropped.
VII Product put into production and marketed.

Fig. 6--Decision box network

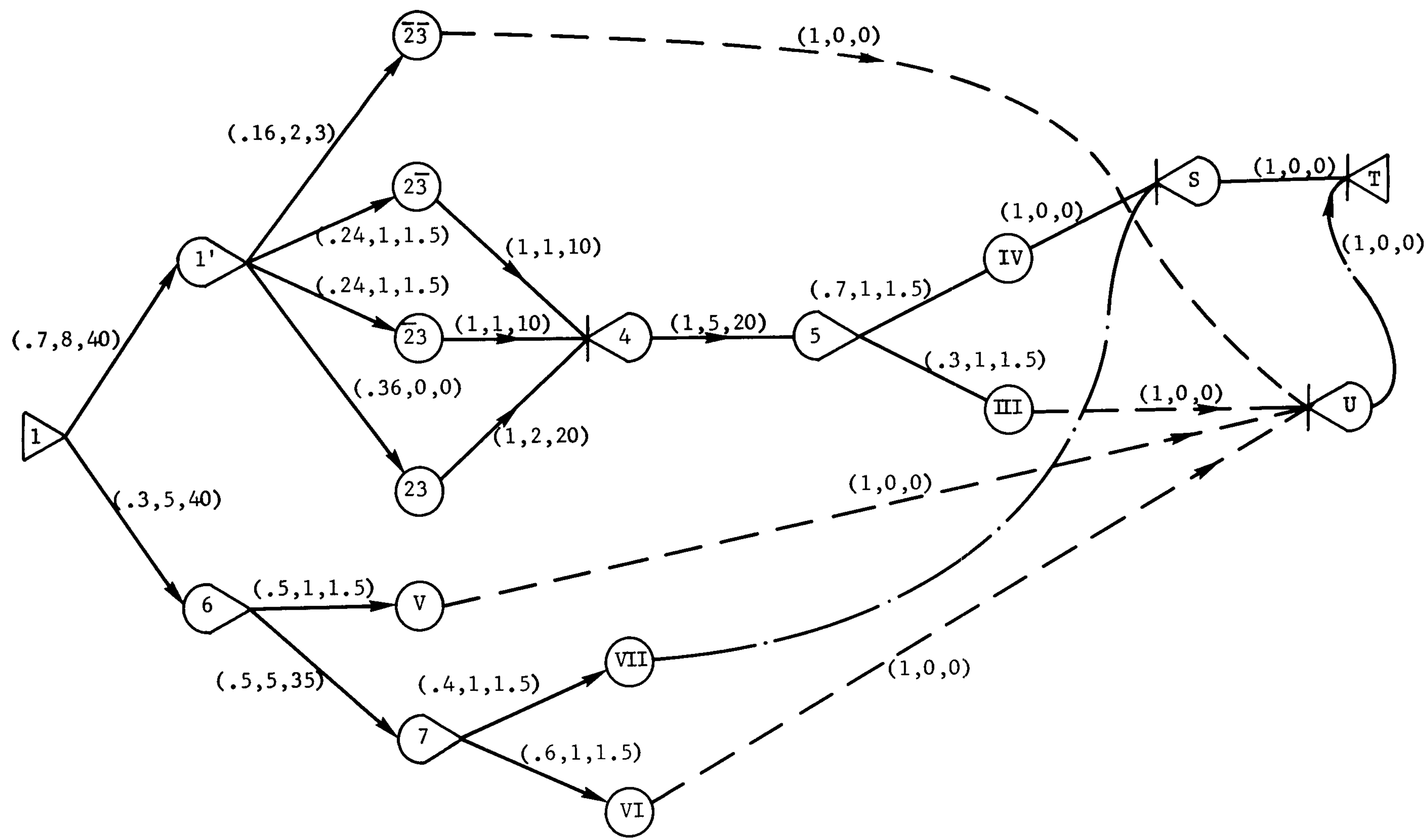
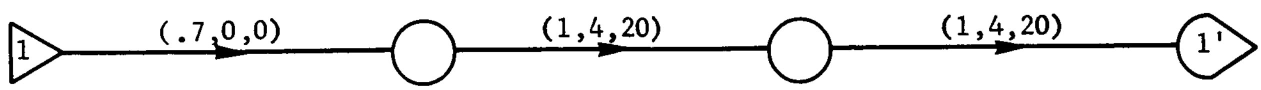
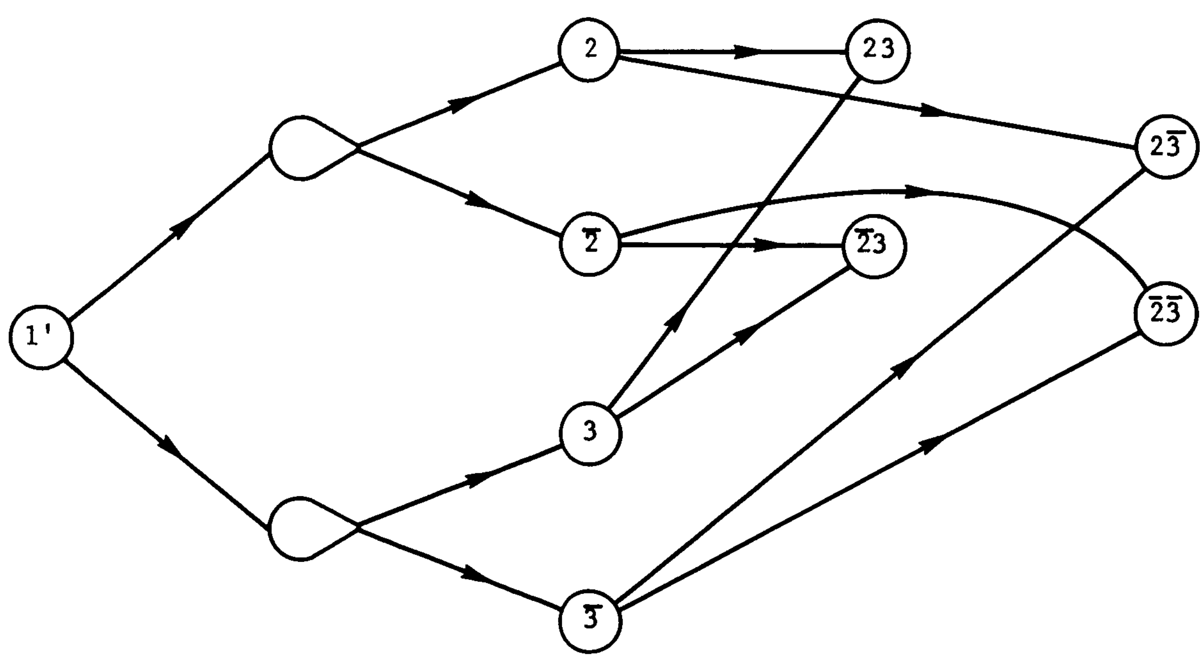


Fig. 7--GERT network of an R&D process

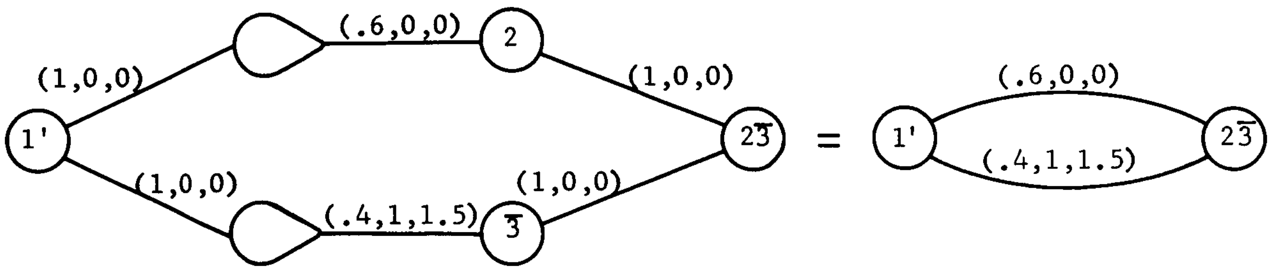
performed simultaneously and should be indicated on the network without the aid of a bracket. The procedure for drawing these specific activities would be:



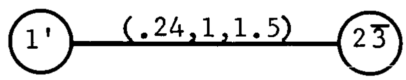
which can be reduced to the branch connecting node 1 to node 1' in Fig. 7. The branches are shown in series because both studies are to be done and the time effort and costs are additive. Second, nodes I and II do not result in the project being dropped as implied in Fig. 6. Also the decision nodes represent specific events, not either-or type of events. For ease of reference between Figs. 6 and 7, nodes have been labeled with two numbers (2 and 3) and the complements of these numbers ($\overline{2}$ and $\overline{3}$). Thus the node, $2\overline{3}$, represents the event AC control has been found to be suitable and DC control has been found to be unsuitable. The detailed segment of the network between node 1' and combinations of nodes 2, $\overline{2}$, 3 and $\overline{3}$ would be:



Considering only the network from 1' to $2\bar{3}$ we have



and



as shown in Fig. 7. Third, three terminal nodes (U, S, and T) have been added. Node U represents the event "project dropped," S represents "project successful," and node T represents the event "project terminated," whether it was successful or not.

The GERT analysis for the network presented in Fig. 7 requires the extension of the w-function to handle two additive parameters. If t and c are independent or information is only desired about them separately, the w-function for a branch becomes $w(s_1,s_2) = pe^{s_1 t + s_2 c}$. For an event of interest, say IV, we have

$$w_{1\text{-}IV}(s_1,s_2) = \left(.7e^{8s_1+40s_2}\right)\left[.24e^{2s_1+11.5s_2} + .24e^{2s_1+11.5s_2} + .36e^{2s_1+20s_2}\right] \cdot \left[\left(e^{5s_1+20s_2}\right)\left(.7e^{s_1+1.5s_2}\right)\right].$$

The performance measures associated with event IV are computed by

$$p_{1\text{-}IV} = w_{1\text{-}IV}(0,0) = (.7)(.84)(.7) = 0.4116,$$

$$E\{t_{1,IV}\} = \frac{\partial}{\partial s_1}\left[\frac{1}{p_{1\text{-}IV}} w_{1\text{-}IV}(s_1,0)\right]_{s_1=0} = 16.00 \text{ weeks},$$

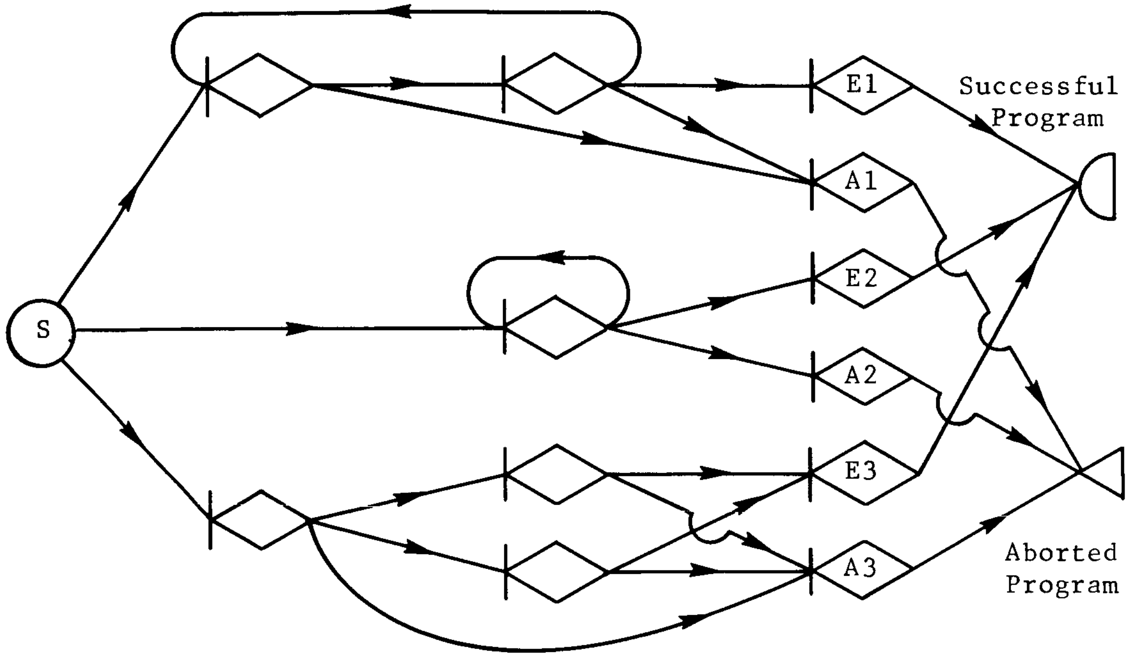
and $$E\{C_{1\text{-}IV}\} = \frac{\partial}{\partial s_2}\left[\frac{1}{p_{1\text{-}IV}} w_{1\text{-}IV}(0,s_2)\right]_{s_2=0} = 76.643 \text{ (in thousands)}.$$

where the expected time and cost are conditioned on the realization of node IV. Higher moments can be calculated by recognizing $M(s_1,s_2) = \frac{1}{p} w(s_1,s_2)$ as the bivariate M.G.F.

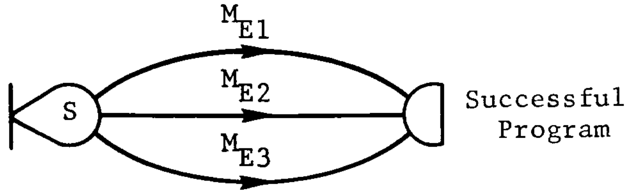
EXAMPLE 13. COSTS ASSOCIATED WITH A LARGE PROGRAM

For a large program there may be independent projects. Associated with the projects are both time and costs, which may be related. If a project is successfully completed, we can consider it to lead to one of the following conditions: (1) fulfill the requirements for the program in the area under concern, in which case there will be no continuing project; (2) activate a new project; or (3) produce results which require a reinvestigation of work accomplished on preceding activities of the project. If a project is not successful then it may have to be done over, or a different approach taken, or the entire program aborted.

As an example of this type of problem, consider the stochastic network shown below of three independent projects, all of which must be completed in order for the total program to be a success:



The procedure discussed in this Memorandum can be used to compute the M.G.F.'s of both the time and costs to go from "S" to any of the three project end points (E) and the three project abort points (A). The extension to the node "successful program" requires a procedure for computing the distribution of a maximum of three random variables, as can be seen from the following reduced network:



For a small network this can be attacked via integration methods. For large networks the techniques discussed in Appendix B may be appropriate. A similar situation exists for the unsuccessful completion of the program where there are three branches going from the "S" node to the abort node. In this case any of the branches leading to the abort node will cause the program to be aborted; hence, the term INCLUSIVE-OR. These problems are discussed in detail in Appendix B.

This example is included not as a solution to a problem, but to raise questions of a practical nature. For the network shown it was assumed that aborting project 1 does not automatically abort projects 2 and 3, but that they continue until either their respective end or abort nodes are reached. This is unrealistic. A mechanism for including this project interdependence is required. One possible device would be a switch placed in the network at points where it would be possible to halt a project based on results from other projects. This too is an area for future research.

The examples discussed in this section demonstrate the power and the diversity inherent in GERT.

VII. THE REVIEW PROCESS

The analysis of stochastic networks discussed so far has concentrated on obtaining performance measures for a specified node. These performance measures can be calculated directly from information concerning the branches. In a review process it is desirable to be able to make statements regarding: (1) confidence limits associated with the system performance measures; and (2) sensitivity of the system performance measures to changes in parameters of individual branches.

CONFIDENCE STATEMENTS

The primary outputs of a GERT analysis are the probability of realizing a node, and the M.G.F. of the time to realize a node given that it is realized. A typical question which might be asked is "What is the probability of realizing a node in T time units?" The answer to this question involves the joint occurrence of realizing the node and the time to realize the node in less than T time units. Symbolically this can be written as

$$P\ (AB) = P(A)\ P(B|A)$$

where A denotes the event "node is realized" and B denotes the event "time of realization of node in less than T time unit."

The quantity P(A) is obtained directly from the GERT analysis. The quantity P(B|A) must be derived from M.G.F. obtained from the GERT analysis. A clarification in terminology is required here: the M.G.F. obtained from the GERT analysis is really a conditional M.G.F., since it is conditioned on the realization of a specific node; thus to obtain P(B|A) it is necessary to obtain the distribution function associated with the derived M.G.F. This problem is referred to in the literature as the inverse transform problem.

The most common inversion method is a table look-up operation. For GERT problems this does not appear practical, due to the complexity of the M.G.F. derived.

A second inversion method is to use an inversion formula. For systems involving only constant times, the inversion formula can be applied by inspection, since all terms are of the form pe^{st}, and the density function is described by all pairs of p and t. For more complex expressions it may be more appropriate to employ the Laplace transform or the characteristic function in the GERT analysis. For these transforms, complex inversion formulii exist. Discussion of the inverse Laplace transform is given in Appendix B.

A third inversion method is to calculate the first n moments of the distribution function from the M.G.F. These moments (the number of moments used depends on the technique employed and the accuracy desired) can then be used to approximate the distribution function. The two most widely used techniques for this approximation are Pearson's curves and Gram-Charlier series. These techniques are discussed in the literature[22,23] and will not be presented in this Memorandum.

As an alternative approach to making confidence statements, the form of the distribution might be assumed. Then confidence statements can be made, using the assumed distribution function. For example, the distribution of the time to reach a node could be assumed to be normal, and with knowledge of the mean and variance, confidence statements can be made. The appropriateness of assuming a distribution form is dependent on the specific problem under study.

ELASTICITY AND SENSITIVITY ANALYSIS

Elasticity is defined as the ratio of the fractional change in one variable to the fractional change in another variable*. The symbol $\mathcal{E}_y^x$ will be used to denote the elasticity of x with respect to changes in y, which is by definition

$$\mathcal{E}_y^x = \frac{dx/x}{dy/_y} . \tag{8}$$

Sensitivity, S, as used here is the rate of change of x with respect to y, <u>viz.</u>,

$$S_y^x = \frac{dx}{dy} . \tag{9}$$

Examples of the x-variable in the above are p_E, μ_{1E}, μ_{2E}, $\cdots$, μ_{nE}, whereas the y-variable might be p_j, μ_{1j}, or μ_{2j} for all j contained in the network.

Sensitivity is an important concept in network evaluation, review, and improvement. The calculation of performance measures of a network as a function of the components of the network has been described as the evaluation portion of GERT. A sensitivity analysis details the changes in the performance measures as a function of changes in the components of a network and, hence, is part of the review procedure. The sensitivity function can be used in the following decision-making areas:

1. Determination of the branch that will most affect the performance measure (if possible added dollars could be expended to affect the change);

2. Determination of branches that should be deleted if a time schedule had to be made; and

3. Determination of the next set of branches to be traversed if scheduling of branches is permissible (this corresponds to an adaptive scheduling procedure).

---

*This is the usual definition of the elasticity of a function. In electrical engineering publications, however, it is referred to as a sensitivity function.

By developing a sensitivity function associated with each branch, it may be possible to assign a criticality index to each branch[24]. Thus the sensitivity function would provide input information for decision-making; the actual decision-making process is outside the scope of GERT.

Since GERT provides a procedure for obtaining expressions for p_E, μ_{1E}, μ_{2E}, etc., as a function of p_j, μ_{1j}, μ_{2j} $\cdots$, the elasticity and sensitivity functions can be obtained directly through partial differentiation. The computation of these review functions for Example 8, The Thief of Bagdad, is given below in Example 14.

EXAMPLE 14. REVIEW PROCESS FOR THE THIEF OF BAGDAD EXAMPLE

From Example 8, the equivalent w-function was derived as

$$w_F(s) = \frac{p_F\ M_F(s)}{1 - p_S\ M_S(s) - p_L\ M_L(s)} \quad .$$

For this example $p_E = 1$ and, hence, $\frac{\partial p_E}{\partial p_j} = 0$ for all j. In general when determining the sensitivity of p_E with respect to a p_j, the interdependence between p_i and p_j must be included. For this example, we have $p_S + p_L + p_F = 1$. Consider now $\mu_{1E} = \frac{d}{ds}\ [M_E(s)]_{s=o}$ from which we have (since $M_E(s) = w_E(s)$ in this case)

$$\mu_{1E} = \frac{p_F\ \mu_{1F}}{1 - p_S - p_L} + \frac{p_F(p_S\ \mu_{1S} + p_L\ \mu_{1L})}{(1 - p_S - p_2)^2} \quad ,$$

which upon simplifying yields

$$\mu_{1E} = \frac{p_F\ \mu_{1F} + p_S\ \mu_{1S} + p_L\ \mu_{1L}}{p_F} \quad .$$

From this equation the following table can be obtained

Review Measure \ j=	F	S	L
$\frac{\partial\mu_{1E}}{\partial p_j}$	$\frac{\mu_{1S}}{p_F}\left[\frac{\partial p_S}{\partial p_F} - \frac{p_S}{p_F}\right] + \frac{\mu_{1L}}{p_F}\left[\frac{\partial p_L}{\partial p_F} - \frac{p_L}{p_F}\right]$	$\frac{\mu_{1S}}{p_F}\left[1 - \frac{p_S}{p_F}\frac{\partial p_F}{\partial p_S}\right] + \frac{\mu_{1L}}{p_F}\left[\frac{\partial p_L}{\partial p_S} - \frac{p_L}{p_E}\frac{\partial p_F}{\partial p_S}\right]$	$\frac{\mu_{1S}}{p_F}\left[\frac{\partial p_S}{\partial p_L} - \frac{p_S}{p_F}\frac{\partial p_F}{\partial p_L}\right] + \frac{\mu_{1L}}{p_F}\left[1 - \frac{p_L}{p_F}\frac{\partial p_F}{\partial p_L}\right]$
$\frac{\partial\mu_{1E}}{\partial\mu_{1j}}$	1	$\frac{p_S}{p_F}$	$\frac{p_L}{p_F}$
$\mathcal{E}^{\mu_{1E}}_{\mu_{1j}}$	$\frac{\mu_{1F}}{\mu_{1E}}$	$\frac{p_S\mu_{1S}}{p_F\mu_{1E}}$	$\frac{p_L\mu_{1L}}{p_F\mu_{1L}}$

The equation for the second moment about zero is

$$\mu_{2E} = \mu_{2F} - 2\,\frac{\mu_{1F}}{p_F}\,[p_S\mu_{1S} + p_L\mu_{1L}] - \frac{1}{p_F}\,[p_S\mu_{2S} + p_L\mu_{2L}]$$

$$+ \frac{1}{p_F^2}\,[p_S\mu_{1S} + p_L\mu_{1L}]^2 \;.$$

The sensitivity and elasticity functions can be computed from Eqs. 8 and 9. A more efficient method for obtaining these functions is needed if their use with large networks is to become computationally practical.

Suppose that the times to traverse any of the tunnels is a constant amount. Thus,

$$M_F(s) = e^{st_F}\,,$$

$$M_S(s) = e^{st_S}\,,$$

$$M_L(s) = e^{st_L}\,,$$

and $M_E(s) = w_E(s) = \dfrac{p_F e^{st_F}}{1 - p_S e^{st_S} - p_L e^{st_L}}\;.$

By proper selection of s it is possible to make $p_S e^{st_S} + p_L e^{st_L} < 1$, and the denominator can be expanded in a power series, i.e.,

$$\frac{1}{1 - p_S e^{st_S} - p_L e^{st_L}} = \sum_{k=o}^{\infty} \left(p_S e^{st_S} + p_L e^{st_L}\right)^k .$$

In the above expression, k represents the number of feedback loops taken (the number of times a nonfreedom tunnel is selected). The binomial expansion can now be employed to yield

$$\left(p_S e^{st_S} + p_L e^{st_L}\right)^k = \sum_{j=o}^{k} \binom{k}{j} \left(p_S e^{st_S}\right)^{(k-j)} \left(p_L e^{st_L}\right)^j .$$

Substituting the above into the equation for $M_E(s)$ yields

$$M_E(s) = p_F e^{st_F} \sum_{k=o}^{\infty} \sum_{j=o}^{\infty} \binom{k}{j} \left(p_S e^{st_S}\right)^{(k-j)} \left(p_L e^{st_L}\right)^j .$$

The density function (from which the distribution function is easily calculated) can be obtained by inverting $M_E(s)$ one term at a time. This involves the enumeration of all terms of $M_E(s)$ by specifying values for k and j. Thus,

$$\text{Prob}\left\{t = t_F + (k - j)t_S + jt_L\right\} = p_F \binom{k}{j} p_S^{(k-j)} p_L^{\,j}$$

$$\text{for } k = 0, 1, 2, \ldots, \text{ and } j \leq k.$$

For practical application the computations could stop when the probabilities computed sum to a desired amount, such as 0.990. Table 2 presents the computations for the Thief of Bagdad problem for $p_F = 0.7$, $p_S = 0.2$, and $p_L = 0.1$.

Table 2

CALCULATION OF PARTIAL DENSITY FUNCTION FOR THIEF OF BAGDAD PROBLEM WITH

$p_F = 0.7$, $p_S = 0.2$, and $p_L = 0.1$, and

$t_F = 0$, $t_S = 1$, and $t_L = 3$.

k	j	t = k + 2j	$P\{t = k + 2j\} = (.7)\binom{k}{j}(.2)^{k-j}(.1)^j$
0	0	0	0.7000
1	0	1	0.1400
1	1	3	0.0700
2	0	2	0.0280
2	1	4	0.0280
2	2	6	0.0070
3	0	3	0.0056
3	1	5	0.0084
4	0	4	0.0011
4	1	6	0.0023
			Total 0.9904

With the above information, sophisticated decision making can proceed. Decisions regarding the efficiency of a network, criticality of specific nodes and branches, and comparison of networks are all subject to investigation by the procedures described.

VIII. AREAS FOR FUTURE RESEARCH

The capacity for growth within GERT is large. Throughout this Memorandum avenues for future research have been discussed. In this section a summary list of the areas for future research is given.

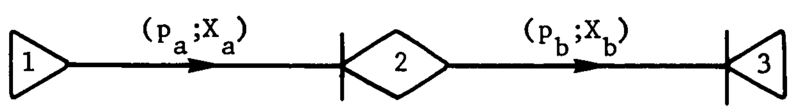
Analysis of other logical operations. This area of future research is discussed in detail in Appendix B. Of major interest is the development of approximation techniques for AND and INCLUSIVE-OR nodes. As more familiarity is gained with GERT, the need for other logical operations such as minimum and invertor operations will become apparent. Further research on the analysis and development of other logical operations would be worthwhile.

Computation programs. A program exists for determining the elements of the w-function of a network and for computing: (1) the probability that a terminal node will be realized; (2) the mean and variance of the time to traverse the network, given the node of interest is realized. There is a need to extend the program to obtain higher moments, especially if confidence statements are desired.

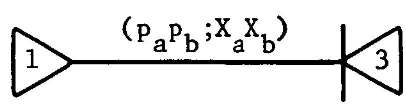
Along with this, further research on methods for making confidence statements, although not peculiar to GERT, would enhance the usefulness of the final output of GERT.

Time-counter interdependence. For stochastic processes it is desirable to be able to determine the distribution of the number of counts of an event at a particular time, or to determine the distribution of the time to obtain a particular number of counts. This research area would require that both the time variable and the count variable be included on the network using a bivariate M.G.F. Difficulties arise in the characterization of the dependence of the two random variables. Initial research on the counter-type operation has been performed.[17]

Multiplicative variables. By multiplicative variables is meant that the variables associated with the branches are multiplied, not added. For example



would be equivalent to



Extension and application of the material presented in Appendix C appears to be a fruitful area for future research.

Review procedure. More efficient methods for obtaining the review functions are needed. Also the use of the review functions for system improvement and optimization is a large area for future research.

Applications of GERT. The best method for a technique to become accepted is through application. The examples presented in this Memorandum are hypothetical; the next step is the application of GERT to practical problems. One such application currently under development is analysis of a space vehicle countdown. Other areas that have been shown to be amenable to analysis using GERT include inventory problems and queueing problems.[17]

Applying and extending GERT in the solution of problems in the above research areas will increase the potentialities of GERT, and hence of the systems analyst.

Appendix A

A COMPUTER PROGRAM FOR ANALYZING GERT NETWORKS

A digital computer program has been written in the FORTRAN IV language to obtain pertinent information for networks with EXCLUSIVE-OR nodes. The program can be used to determine the source nodes, the sink nodes, the paths connecting the source nodes to the sink nodes, and the loops of a network. The output from the program includes: (1) the paths and loops of a network; (2) the probability of realizing a sink node from any source node and; (3) 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 program accepts as its input the branches of the network, as described by the nodes from whence its originates and to where it terminates. Associated with each branch is a probability and a M.G.F. The M.G.F. is described by a three-letter code and by appropriate parameters of interest. The program based on this input information determines all paths and loops of the network. From the values associated with the paths and loops of the network the desired output statistics can be computed.

In this appendix the method for calculating the output statistics, the operating procedure for the program, and sample problems will be presented.

CALCULATION OF NETWORK STATISTICS

The program accepts the input information and determines the source and sink nodes and all paths and loops of the network. In addition the program determines the following three values associated with a path or loop:

1. probability of traversal;
2. the mean time of traversal; and
3. the second moment of the time of traversal.

These values are determined by the following methods. The w-function associated with a path or loop is the product of the w-functions of the branches which make up the path or loop. Letting L represent a path or loop, we have

$$w_L(s) = \prod_{i \in L} w_i(s).$$

Now the probability associated with L is

$$p_L = w_L(o) = \prod_{i \in L} w_i(o) = \prod_{i \in L} p_i .$$

The expected time to traverse L is given by

$$\mu_{1L} = \frac{1}{w_L(o)} \left.\frac{\partial w_L(s)}{\partial s}\right|_{s=o} = \frac{1}{w_L(o)} \left(\prod_{i \in L} w_i(s)\right)\left(\sum_{i \in L} \frac{1}{w_i(s)} \frac{\partial w_i(s)}{\partial s}\right)_{s=o}$$

$$\mu_{1L} = \frac{1}{w_L(o)} \left(\prod_{i \in L} w_i(o)\right)\left(\sum_{i \in L} \left.\frac{\partial M_i(s)}{\partial s}\right|_{s=o}\right) = \sum_{i \in L} \mu_{1i} .$$

The above says that the expected time to traverse a path or loop is the sum of the expected times of the branches of the path or loop. The complex analysis is given to lay the foundation for obtaining an equation for the second moment. From the w-function we have

$$\mu_{2L} = \frac{1}{w_L(o)} \left. \frac{\partial^2 w_L(s)}{\partial s^2} \right|_{s\to o}$$

$$= \frac{1}{w_L(o)} \left(\prod_{i\epsilon L} w_i(s) \right) \left\{ \left[\sum_{i\epsilon L} \frac{1}{w_i(s)} \frac{\partial w_i(s)}{\partial s} \right]^2 - \sum_{i\epsilon E} \left(\frac{1}{w_i(s)} \frac{\partial w_i(s)}{\partial s} \right)^2 \right.$$

$$\left. + \sum_{i\epsilon L} \frac{1}{w_i(s)} \frac{\partial^2 w_i(s)}{\partial s^2} \right\}_{s\to o}$$

$$= \mu_{1L}^2 - \sum_{i\epsilon L} \mu_{1i}^2 + \sum_{i\epsilon L} \mu_{2i} \quad .$$

The computer program computes p_L, μ_{1L}, and μ_{2L} for all paths and loops of the network, including loops which are products of disjoint loops. These values are then combined through the topological equation to obtain the output statistics desired. The equivalent w-function for one path, A, between the two nodes of interest is given by:

$$w_E(s) = \frac{A(s) \left[1 + \sum_{i=1}^{\infty} (-1)^i \sum_{k=1}^{n_i} w_{L_k}^{(i)}(s) \right]}{\left[1 + \sum_{j=1}^{\infty} (-1)^j \sum_{v=1}^{n_j} w_{L_v}^{(j)}(s) \right]} = \frac{A(s)\ B(s)}{D(s)} = \frac{N(s)}{D(s)}$$

where $A(s) \equiv$ product of the values of all branches in the path considered;

$w_{L_k}^{(i)}(s)$ = product of the values of i disjoint loops having no nodes in common with path A;

n_i = the number of loops composed of i disjoint loops;

$w_{L_v}^{(j)}(s)$ = product of the values of any j disjoint loops;

n_j = the number of loops composed of j disjoint loops;

and B(s), D(s) and N(s) are direct substitutions. If there is more than one path, then the w-functions associated with each path would be summed. For convenience, consider the one path case. The output statistics can be computed from the following equations:

$$p_E = w_E(o)$$

$$\mu_{1E} = \frac{1}{w_E(o)} \left. \frac{\partial w_E(s)}{\partial s} \right|_{s=o} = \frac{1}{w_E(o)} \left[\frac{D(s)\frac{\partial N(s)}{\partial s} - N(s)\frac{\partial D(s)}{\partial s}}{D^2(s)} \right]_{s=o}$$

$$\mu_{2E} = \frac{1}{w_E(o)} \left. \frac{\partial w_E^2(s)}{\partial s^2} \right|_{s=o}$$

$$= \frac{1}{w_E(o)} \left\{ \frac{D(s)\left[D(s)\frac{\partial^2 N(s)}{\partial s} - N(s)\frac{\partial^2 D(s)}{\partial s^2}\right] - 2\frac{\partial D(s)}{\partial s}\left[D(s)\frac{\partial N(s)}{\partial s} - N(s)\frac{\partial D}{\partial s}\right]}{D^3(s)} \right\}_{s=o}$$

and $\sigma_E^2 = \mu_{2E} - \mu_{1E}^2$.

In the above equations the values of $\frac{\partial N(s)}{\partial s}$, $\frac{\partial^2 N(s)}{\partial s^2}$, etc., evaluated at s=o, are obtained from the previously compiled values of μ_{1L}, μ_{2L}, etc. For example

$$\left. \frac{\partial N(s)}{\partial s} \right|_{s=o} = \mu_{1A} \left[1 + \sum_{i=1}^{\infty} (-1)^i \sum_{k=1}^{n_i} \mu_{1L_k}^{(i)} \right] .$$

The above equations are included in the computer program described in this appendix.

PROGRAM OPERATING PROCEDURE

The GERT program is written in FORTRAN IV. The program has been debugged and run on the IBM 7040-44 and the CDC 3400. The input specifications to the program are given in Table 3. The equations and moments of the distributions that have been programmed are presented in Table 4. Other distributions can be handled by this

Table 3

INPUT TO GERT PROGRAM

Each network is specified by defining its links as follows:

1. Node beginning the link
2. Node terminating the link
3. Type of distribution of time associated with the link
4. Probability of utilization of the link if at its beginning node
5. Coefficients defining the time distribution

Field 1 (cc. 1-4) = Node beginning link (right justified)

Field 2 (cc. 6-9) = Node terminating link (right justified)

Field 3 (cc. 11-13) = Type of distribution [B, D, E, GA, GE, NB, NØ, P, U] (left justified)

Field 4 (cc. 14-20)
Field 5 (cc. 21-27)
Field 6 (cc. 28-34)
Field 7 (cc. 35-41)
Field 8 (cc. 42-48)
Field 9 (cc. 49-55)
Field 10 (cc. 56-62)
Field 11 (cc. 63-69)

See table below for definitions of these fields. The format for all fields is F7.3.

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	-	-	-	-	-
NØ (Normal	Prob.	m	σ	-	-	-	-	-
P (Poisson)	Prob.	λ	-	-	-	-	-	-
U (Uniform	Prob.	a	b	-	-	-	-	-

Each deck of cards defining a network must be followed by a blank card (Field 1=0 or blank).

Table 4

DISTRIBUTIONS 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)	$\frac{p_1e^{st_1}+p_2e^{st_2}+\dots}{p_1+p_2+\dots}$	$\frac{p_1T_1+p_2T_2+\dots}{p_1+p_2+\dots}$	$\frac{p_1T_1^2+p_2T_2^2+\dots}{p_1+p_2+\dots}$	$w_E(o);p_1,T_1,p_2,T_2;$
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)	$\left(\frac{p}{1-e^s+pe^s}\right)^r$	$\frac{r(1-p)}{p}$	$\frac{r(1-p)(1+r-rp)}{p^2}$	$w_E(o);r,p$
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)	$\frac{e^{sa}-e^{sb}}{(a-b)s}$	$\frac{a+b}{2}$	$\frac{a^2+ab+b^2}{3}$	$w_E(o);a,b$

program by specifying the normal distribution with mean and standard deviation values of the distribution of interest. This follows from the observation that the n^{th} moment of the equivalent function is dependent only on the n^{th} or smaller moments of the branch functions. Thus, $\mu_{1E} = f(\mu_{ij})$ and $\mu_{2E} = f(\mu_{2j}, \mu_{1j})$. The equations of the previous section illustrate that μ_{2E} is not a function of the distribution form but only the values associated with μ_{1j} and μ_{2j}.

A sample of the output will now be described. First, an echo check of the input network is given as shown in Table 5.

Table 5

GERT PROGRAM OUTPUT: ECHO CHECK OF INPUT NETWORK

Input Network										
1	3 D	1.000	0.	-0.	-0.	-0.	-0.	-0.	-0.
2	4 D	1.000	0.	-0.	-0.	-0.	-0.	-0.	-0.
3	5 D	0.600	2.000	-0.	-0.	-0.	-0.	-0.	-0.
3	6 D	0.400	4.000	-0.	-0.	-0.	-0.	-0.	-0.
4	5 D	0.300	5.000	-0.	-0.	-0.	-0.	-0.	-0.
4	6 D	0.700	2.000	-0.	-0.	-0.	-0.	-0.	-0.
5	3 D	0.200	7.000	-0.	-0.	-0.	-0.	-0.	-0.
6	4 D	0.300	6.000	-0.	-0.	-0.	-0.	-0.	-0.
5	7 D	0.800	0.	-0.	-0.	-0.	-0.	-0.	-0.
6	8 D	0.700	0.	-0.	-0.	-0.	-0.	-0.	-0.

The network corresponding to this input information is given in Fig. 8.

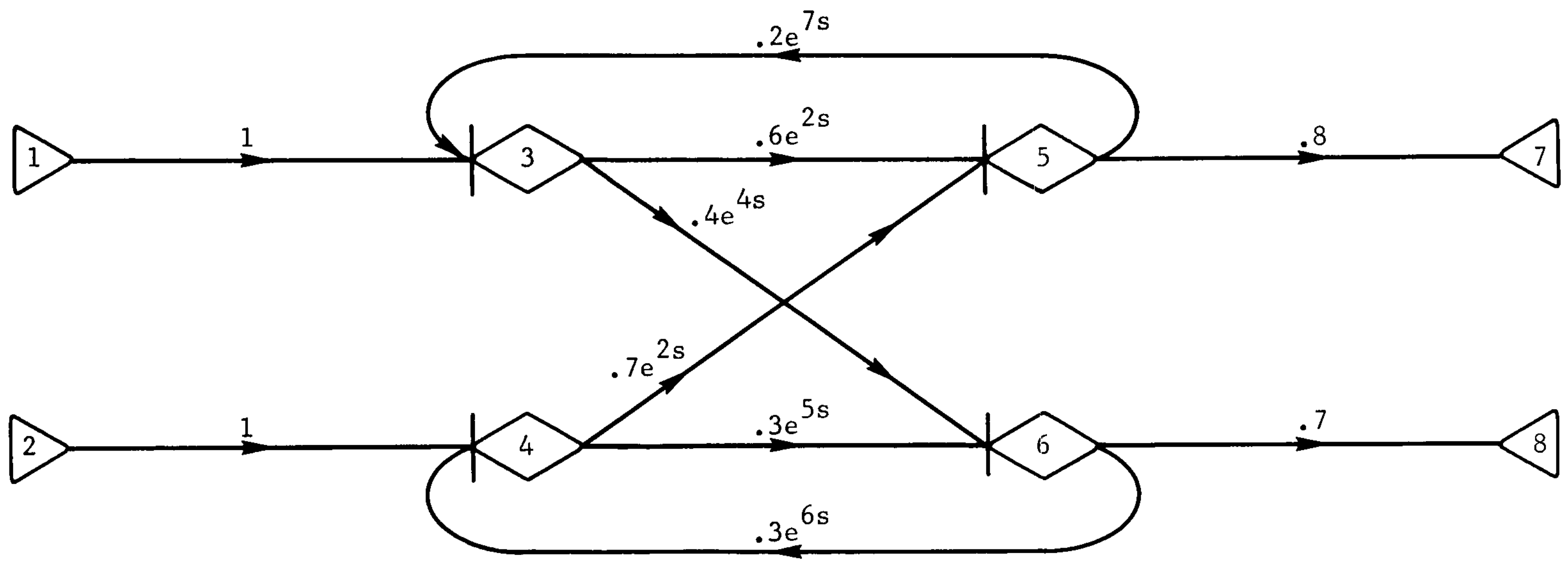


Fig. 8--Sample problem GERT network corresponding to Table 5

Intermediate output available from the program is shown in Tables 6 through 8. The final results are given in Table 9.

Table 6

CALCULATION OF BRANCH PARAMETERS

Input Network

Nodes and probability of selection with mean and variance of time for each link.

From	To	Probability	Mean	Variance
1	3	1.000	0.	-0.
2	4	1.000	0.	-0.
3	5	0.600	2.000	-0.
3	6	0.400	4.000	-0.
4	5	0.300	5.000	0.000
4	6	0.700	2.000	-0.
5	3	0.200	7.000	-0.
6	4	0.300	6.000	0.000
5	7	0.800	0.	-0.
6	8	0.700	0.	-0.

Table 7

LISTING OF NETWORK LOOPS

Loop of Order 1	w(0) = 0.120000					
w(0) = 0.1200 .	Nodes	3	5			
Loop of Order 2	w(0) = 0.025200					
w(0) = 0.1200 .	Nodes	3	5			
w(0) = 0.2100 .	Nodes	4	6			
Loop of Order 1	w(0) = 0.007200					
w(0) = 0.0072 .	Nodes	3	6	4	5	
Loop of Order 1	w(0) = 0.210000					
w(0) = 0.2100 .	Nodes	4	6			

Table 8

LISTING OF NETWORK PATHS

NS	NE	Probability	M T	V T	(Nodes in Path)
1	7	0.551163	3.4926	19.7059	1 3 5 7
1	7	0.041860	18.6192	41.2410	1 3 6 4 5 7
1	8	0.406977	7.6192	41.2409	1 3 6 8
2	8	0.024419	19.6192	41.2410	2 4 5 3 6 8
2	7	0.348837	8.6192	41.2410	2 4 5 7
2	8	0.626744	4.3919	28.6893	2 4 6 8

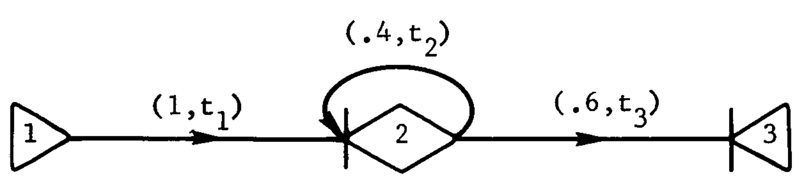
Table 9

EQUIVALENT BRANCHES OF THE NETWORK

Entry	Exit	Probability	Mean Time	Variance Time
1	7	0.593023	4.5604	36.2375
1	8	0.406977	7.6192	41.2409
2	8	0.651163	4.9629	37.5290
2	7	0.348837	8.6192	41.2410

SOLUTIONS OF STOCHASTIC PROCESSES AND THREE-PLAYER GAME PROBLEMS USING GERT PROGRAM

In Sec. V of this Memorandum, an analysis of a random number of random variables was presented. The GERT network for a problem of this type is



This network was analyzed for four different distributions each having the same mean. The results are given in Table 10.

Table 10

GERT PROGRAM OUTPUT OF A NETWORK INVOLVING A SUM OF A RANDOM NUMBER OF RANDOM VARIABLES

Distribution type	μ_{11}	μ_{21}	μ_{12}	μ_{22}	μ_{13}	μ_{23}	μ_{1E}	Var_E
Constant	5	0	2	0	3	0	9.3333	4.4444
Exponential	5	25	2	4	3	9	9.3333	41.1113
Poisson	5	5	2	2	3	3	9.3333	13.7778
Normal	5	3	2	3	3	3	9.3333	28.4444

The first moments of the equivalent network for each distribution are the same, since all the first moments of the individual branches are the same. Note, however, the increase in the variance from the discrete case to the exponential case. This increase is due to the different values associated with the variance of the individual branches. The variance in the constant case (which is a form of discrete distribution) is due entirely to the uncertainty in branch selection.

As a second example, the GERT program was used to analyze the three-player game given in Example 9, in which the game is won by the first player to win two consecutive games. The network is shown in Fig. 9. It will be 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 will be assumed to be one for each game played.

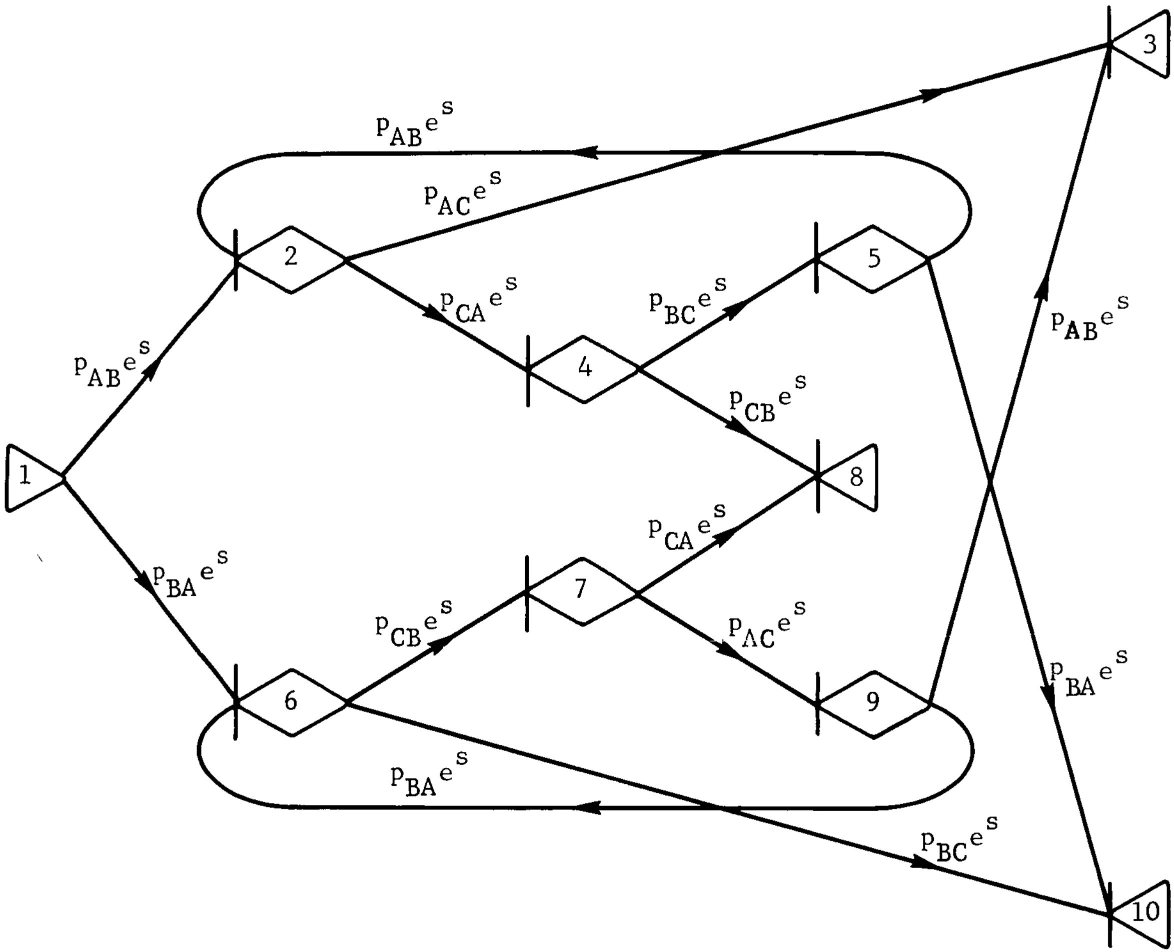


Fig. 9--GERT network of a three-player game

The GERT program was run to analyze the game when the independent variable was $p_{AB} = p_{BC} = p_{CA}$. The results of the program are given in Table 11. Since the game is symmetrical, only values of $p_{AB} \leq .5$ are tabulated.

The results presented in Table 11 show that player A has the highest probability of winning the game for values of p_{AB} in the range 0.10 to 0.50. This is somewhat contrary to intuition. It is also seen that as p_{AB} moves away from the value .50, the expected number and the variance of the games played increases until a winner is declared.

Table 11

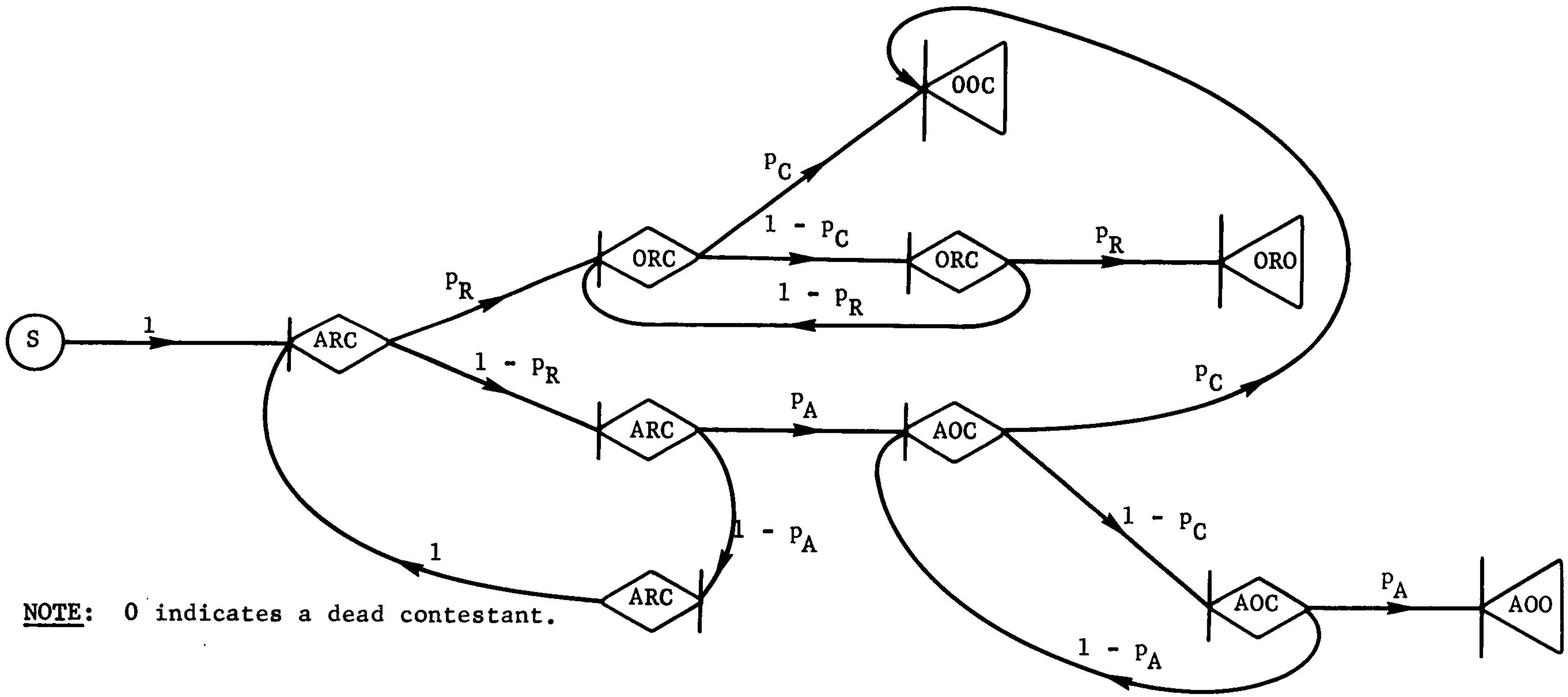
RESULTS OF THREE-PLAYER GAME WITH $p_{AB} = p_{BC} = p_{CA}$

	A Wins			B Wins			C Wins		
p_{AB}	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	4.9210	17.4220	0.3343	5.1259	19.0015	0.2946	5.8055	18.1899
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	0.3666	2.9931	2.8553	0.3471	2.9895	3.0647	0.2862	3.6039	2.3538
0.45	0.3621	2.8707	2.2638	0.3521	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

The above examples illustrate the results obtained from the current GERT program. The examples were chosen to be simple, yet they do not have simple solutions.

As a test of the reader's intuition, consider the following situation. Three contestants have weapons which they can use against one of their opponents: man A has a probability of p_A of killing an opponent when he uses his weapon; man R has a probability of p_R of killing an opponent when he uses his weapon; and man C has a probability of p_C of killing his opponent when using his weapon. The object of A, R, and C is to defeat the other opponents. Suppose $p_A > p_R > p_C$ and R fires at A first. Then A would return R's fire. C being rational decides he would be better off not to fire until only one opponent is left. Two questions of interest for A are: (1) for what values of p_A, p_R, and p_C should A fire first and; (2) if $p_A = .90$, $p_R = .75$ and $p_C = .40$, what is the probability under the above firing sequence of A winning?

The answer to these questions can be approached through GERT. The network for answering question (2) is given in Fig. 10. Superimposing a time or weapons limit on this problem introduces a second, additive, parameter. The analysis of this problem requires a counter operation in GERT.



NOTE: 0 indicates a dead contestant.

Fig. 10--A three-contestant struggle

Appendix B

OTHER LOGICAL ELEMENTS

The AND and INCLUSIVE-OR nodes introduce into an analysis the complex operations of maximization and minimization of the time parameter. This is in contrast to the EXCLUSIVE-OR node which required a summing or an either-or type operation. Thus it is expected that the AND and INCLUSIVE-OR nodes will present difficult mathematical problems.

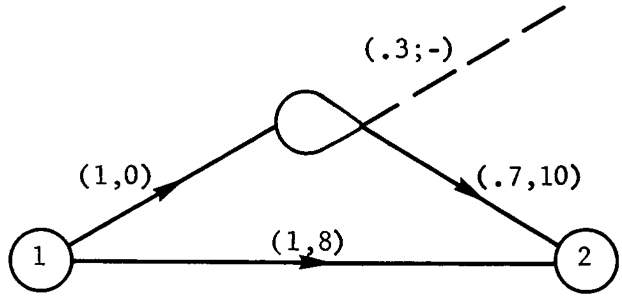
In Sec. I, two simplifying attributes regarding AND and INCLUSIVE-OR nodes were given: (1) the node type does not affect the analysis of two branches in series; and (2) a feedback branch is only meaningful if incident on an EXCLUSIVE-OR node. With this information, the analysis can be directed to the study of parallel branches and series-parallel networks.

Before proceeding, it should be pointed out that no general solution to the AND and INCLUSIVE-OR node analysis has been obtained. The purpose here is to present concepts, approaches, and examples. Where pertinent, approximation possibilities will be indicated. There is a need for future research in this area which this appendix hopefully will instigate.

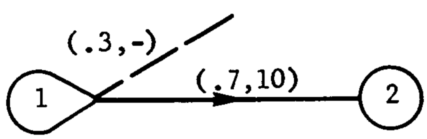
THE AND LOGIC ELEMENT

Three main problems associated with AND nodes are: (1) a semantic problem associated with the probability of realizing a branch leading to an AND node; (2) an analysis based on expected values leads to erroneous results; and (3) the incorporation of the maximum operation is computationally difficult.

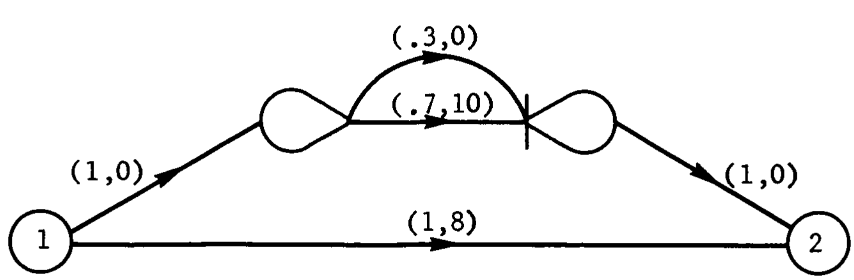
The semantic problem involves the possibility that an AND node will not be realized. Consider the network shown below:



First note that the output of an AND-DETERMINISTIC node must have a p-parameter of 1. This necessitates the use of the AND-PROBABILISTIC node. Now as the network is drawn only 70 per cent of the time will node 2 be realized, since the upper path dominates the lower path, and when it is realized it occurs 10 time units after node 1. This can be represented by

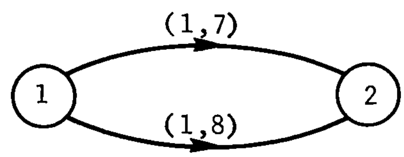


Further, suppose the originator of the network really desired to have the upper path occur 30 per cent of the time without the activity which required 10 time units. This would be drawn as

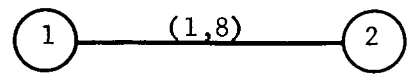


Both networks are feasible; however, they represent different systems.

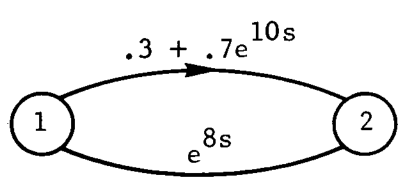
This last network will be used to illustrate the fallacy in reducing a network by use of expected values. For the upper path, the expected time is (.3)(0) + (.7)(10) = 7. The probability of realizing node 2 on the upper path is one, hence we appear to have



which reduces to



The above states that the time to realize node 2 is eight time units after the realization of node 1. Review of the original network shows that 70 per cent of the time the equivalent time is 10 time units and 30 per cent of the time it is eight time units. Analyzing this network with the w-function, we have



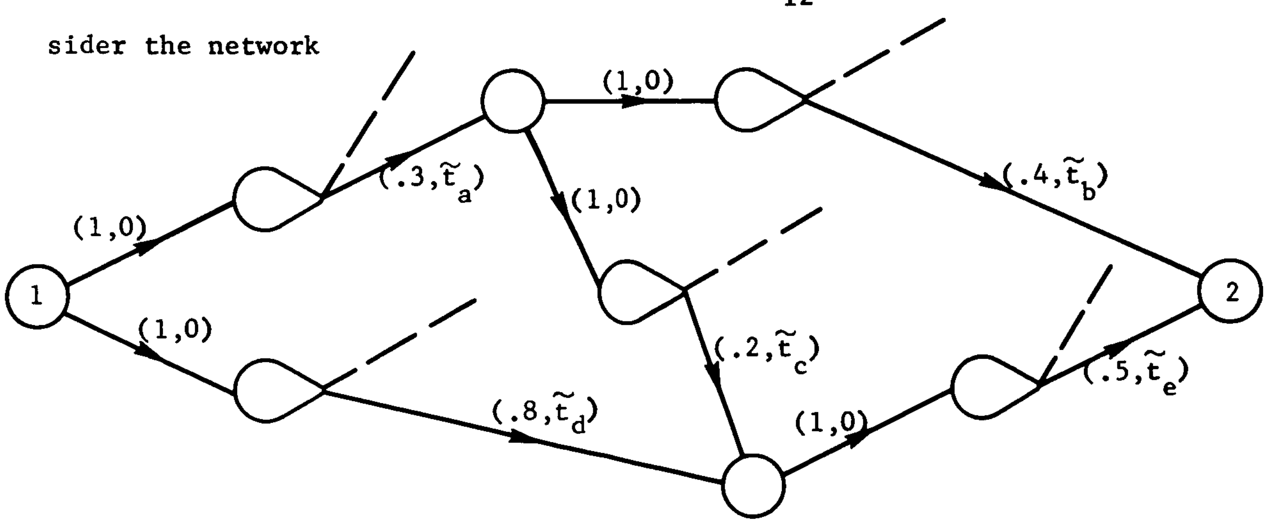
.

By the characteristic of the AND node, the probability of realizing node 2 is the intersection of the probabilities of the branches leading into node 2, and the time is the maximum of the times. Procedures for handling these two calculations will now be investigated.

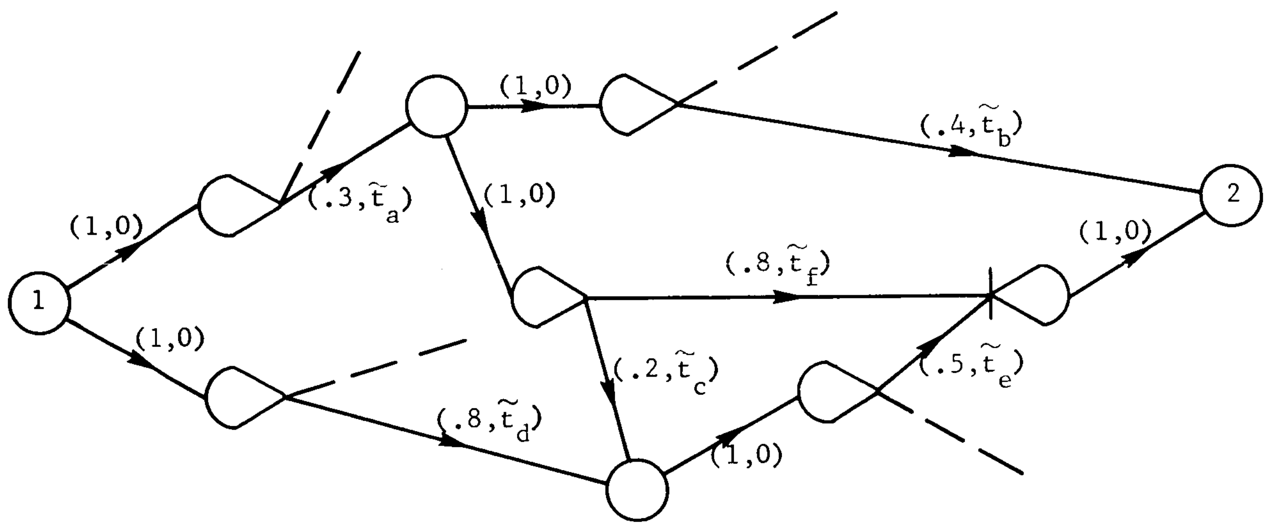
Probability of Realizing an AND Node

This discussion will be limited to the situation in which all probabilities are independent for each branch of a network. Dependence can occur, however, by a branch being on more than one path. There are three distinct possibilities associated with the calculation of probabilities when dealing with AND nodes exclusively: (1) branches in series; (2) branches in parallel with independent probabilities associated with each branch; and (3) paths in parallel with dependent probabilities. For all three cases the probabilities multiply. In case 3 if p_a is included on two parallel paths, then $p_{a \cap a} = p_a$ is the probability calculation. This simple analysis leads to the result that the probability of realizing an AND node in a network consisting only of AND nodes (input side) can be obtained by multiplying the probabilities of each branch of the network leading from the source node to the sink node. Several examples are given below.

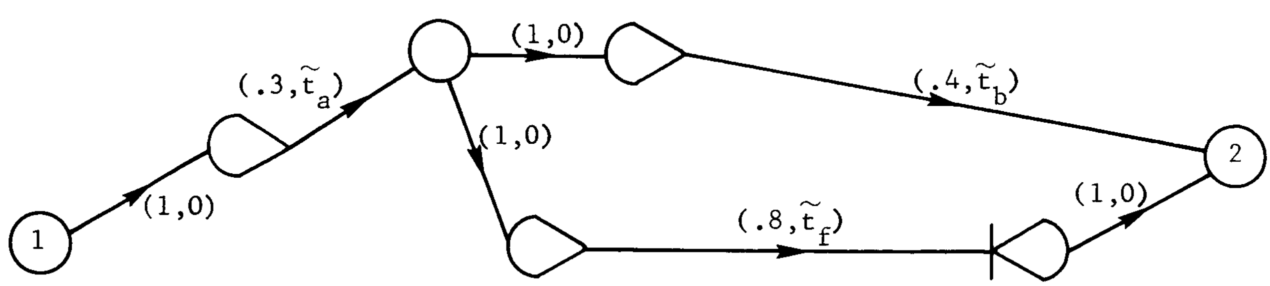
For the situation presented above we obtain the probability of the upper path from w(o) = .3 + .7 = 1.0 and $p_{12} = (1)(1) = 1$. Consider the network



The probability of realizing node 2, p_{12}, is $(.3)(.4)(.2)(.8)(.5) = 0.0096$. The procedure presented involves only the consideration of branches that lead to the final node and the probabilities that those branches are realized. Caution must be taken not to include the probabilities that a given time on a branch is taken. This possibility is eliminated by specifying only AND nodes in the network. If other types of nodes are present, it is necessary to reduce the network. If EXCLUSIVE-OR nodes are present that permit the realization of the final node by more than one set of paths, then the sets of paths can be analyzed independently by the above rule and the probabilities added. For example, if the above network is altered to



then p_{12} = 0.0096 + 0.0960 = 0.1056 where the second factor on the right, (.0960) = (.3)(.4)(.8), is obtained from the network



The time to realize node 2, given it is realized, is dependent on the network that causes node 2 to be realized--not on the individual branch probabilities. This is because in the realization of node 2, all branches of the network must be traversed. The either-or situation defines a different network by which node 2 is caused to be realized. For the above example we require the M.G.F. of $\tilde{t}_{12}^{(1)}$ and $\tilde{t}_{12}^{(2)}$ where

$$\tilde{t}_{12}^{(1)} = \max\,(\tilde{t}_a + \tilde{t}_b;\ \tilde{t}_a + \tilde{t}_c + \tilde{t}_e;\ \tilde{t}_d + \tilde{t}_e)$$

and
$$\tilde{t}_{12}^{(2)} = \max\,(\tilde{t}_a + \tilde{t}_b;\ \tilde{t}_a + \tilde{t}_f)\ .$$

The M.G.F. of the equivalent network would then be

$$M_E(s) = \frac{1}{p_{12}}\left[p_{12}^{(1)}\,M_{12}^{(1)}(s) + p_{12}^{(2)}\,M_{12}^{(2)}(s)\right]$$

where the superscripts refer to the independent networks described previously.

It is now necessary to discuss the calculation of the M.G.F. or, equivalently, the distribution function of the equivalent time parameter.

The Maximum of Random Variables

The preceding discussion shows that the analysis of AND nodes is similar to the analysis of a PERT-type network. To date there is no exact solution to the analysis of PERT networks that is computationally tractible. The purpose of this discussion is to present the actual methods, which for small GERT networks can be applied directly, and which provide background information for future research on approximation methods.

Consider the calculation of the distribution function,

$$F_{12}(t) = \text{Prob}\,(\tilde{t}_{12} \le t)$$

where
$$\tilde{t}_{12} = \max\,(\tilde{t}_a, \tilde{t}_b)\ .$$

For $\tilde{t}_{12} \le t$, then $\tilde{t}_a \le t$ and $\tilde{t}_b \le t$.

The probability of this occurring is

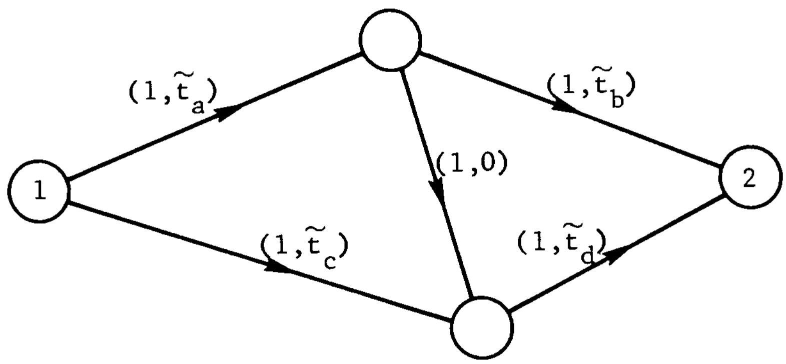
$$\text{Prob }(\tilde{t}_{12} \le t) = \text{Prob }(\tilde{t}_a \le t;\ \tilde{t}_b \le t) \ .$$

If $\tilde{t}_a$ and $\tilde{t}_b$ are independent random variables, then

$$\text{Prob }(\tilde{t}_a \le t;\ t_b \le t) = \text{Prob }(\tilde{t}_a \le t)\ \text{Prob }(\tilde{t}_b \le t) \ ,$$

and we obtain $$F_{12}(t) = F_a(t)\ F_b(t) \ .$$

Consider now the more complex network



in which $\tilde{t}_{12} = \max\ (\tilde{t}_a + \tilde{t}_b;\ \tilde{t}_a + \tilde{t}_d;\ \tilde{t}_c + \tilde{t}_d)$.

Now $$\text{Prob }(\tilde{t}_{12} \le t) = \text{Prob }(\tilde{t}_a + \tilde{t}_b \le t;\ \tilde{t}_a + \tilde{t}_d \le t;\ \tilde{t}_c + \tilde{t}_d \le t)$$

$$= \text{Prob }(\tilde{t}_b \le t - \tilde{t}_a;\ \tilde{t}_d \le t - \tilde{t}_a;\ \tilde{t}_d \le t - \tilde{t}_c) \ .$$

If $\tilde{t}_a \ge \tilde{t}_c$, we have the conditional probability that

$$\text{Prob }(\tilde{t}_{12} \le t | \tilde{t}_a \ge \tilde{t}_c) = \text{Prob }(\tilde{t}_b \le t - \tilde{t}_a;\ \tilde{t}_d \le t - \tilde{t}_a) \ .$$

If $\tilde{t}_a < \tilde{t}_c$, then

$$\text{Prob }(\tilde{t}_{12} \le t | \tilde{t}_a < \tilde{t}_c) = \text{Prob }(\tilde{t}_b \le t - \tilde{t}_a;\ \tilde{t}_d \le t - \tilde{t}_c) \ .$$

By a theorem on total probability we have

$$\text{Prob}\ (\tilde{t}_{12} \le t) = \text{Prob}\ (\tilde{t}_{12} \le t | \tilde{t}_a \ge \tilde{t}_c)\ \text{Prob}\ (\tilde{t}_a \ge \tilde{t}_c)$$

$$+ \text{Prob}\ (\tilde{t}_{12} \le t | \tilde{t}_a < \tilde{t}_c)\ \text{Prob}\ (\tilde{t}_a < \tilde{t}_c)\ .$$

We also have

$$\text{Prob}\ (\tilde{t}_a \ge \tilde{t}_c) = \text{Prob}\ (\tilde{t}_a \ge t;\ \tilde{t}_c \le t) = \text{Prob}\ (t_a \ge t)\ \text{Prob}\ (t_c \le t)$$

$$= [1 - \text{Prob}\ (\tilde{t}_a \le t)]\ \text{Prob}\ (\tilde{t}_c \le t),$$

since $\tilde{t}_a$ and $\tilde{t}_c$ are independent. From this information $F_{12}(t)$ can be calculated.

There are two difficulties with the above approach: (1) for larger networks the calculations could be intractible; and (2) for GERT networks the distribution functions are not known. The first of these difficulties will have to be handled by approximation techniques, which at this point will have to be problem oriented. The second difficulty will now be resolved.

AND Nodes and the s-plane. GERT up to this point has dealt primarily with M.G.F., which enabled the analysis to deal strictly with real variables. For AND nodes it is necessary to perform some analysis using complex variables. This will be done by introducing the Laplace transform in which the variable s is a complex variable. By definition the Laplace transform of a density function $f_k(t)$ is

$$L_k(s) = \int_0^\infty e^{-st}\ f_k(t)\ dt\ .$$

(Note that $L_k(s)$ can be obtained from $M_k(s)$ by substituting -s for s even though the s variables are not the same.) The inverse Laplace transform is defined by the complex integral

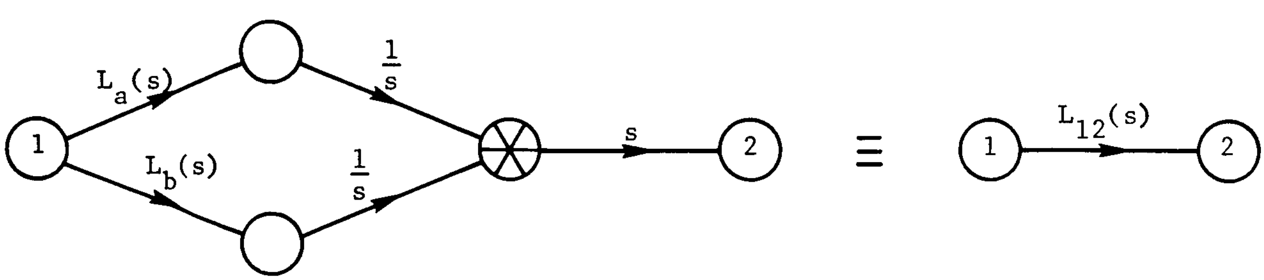
$$f_k(t) = \frac{1}{2\pi j} \int_{\sigma_1 - j\infty}^{\sigma_1 + j\infty} e^{st} L_k(s)\, ds \ .$$

By a theorem in Laplace transforms, the transform of the distribution function for $\tilde{t}_k$ would be $\frac{L_k(s)}{s}$. Further, multiplication of two distribution functions, such as are required by the maximum operation, correspond to a complex convolution of their Laplace transforms. Thus

$$L_{12}(s) = \frac{1}{2\pi j} \int_{\sigma_1 - j\infty}^{\sigma_1 + j\infty} \frac{L_a(q)}{q} \frac{L_b(s - q)}{(s - q)}\, dq \ ,$$

where σ_1 is chosen to separate the singularities of $\frac{L_a(q)}{q}$ and $\frac{L_b(s - q)}{s - q}$.

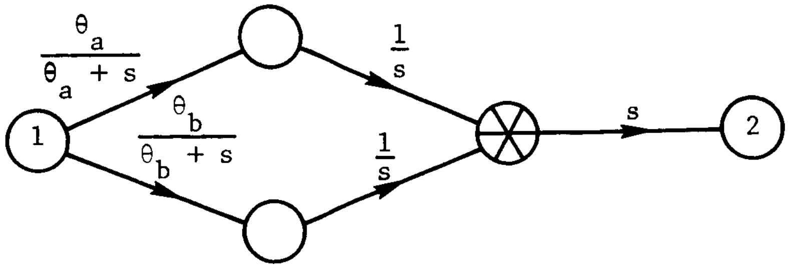
In GERT network terminology we have:



where the symbol ⊗ represents a complex convolution which will be discussed by example. The practicality of this approach hinges on the ability to perform the complex convolution operation. Fortunately in some instances the complex convolution can be replaced by a Bromwich contour integration, which can be accomplished through the use of

Cauchy's residue theorem. Several examples will now be given to demonstrate the approach.

Consider the specific network drawn below:



which can be recognized as the maximum of two exponential distributions. The equation of this network is

$$\frac{1}{s} L_{12}(s) = \int_{\sigma_1 - j\infty}^{\sigma_1 + j\infty} \frac{1}{q}\left(\frac{\theta_a}{\theta_a + q}\right)\left(\frac{1}{s-q}\right)\left(\frac{\theta_b}{\theta_b + s - q}\right) dq$$

where $0 < \sigma_1 < \sigma = \mathrm{Re}(s)$.

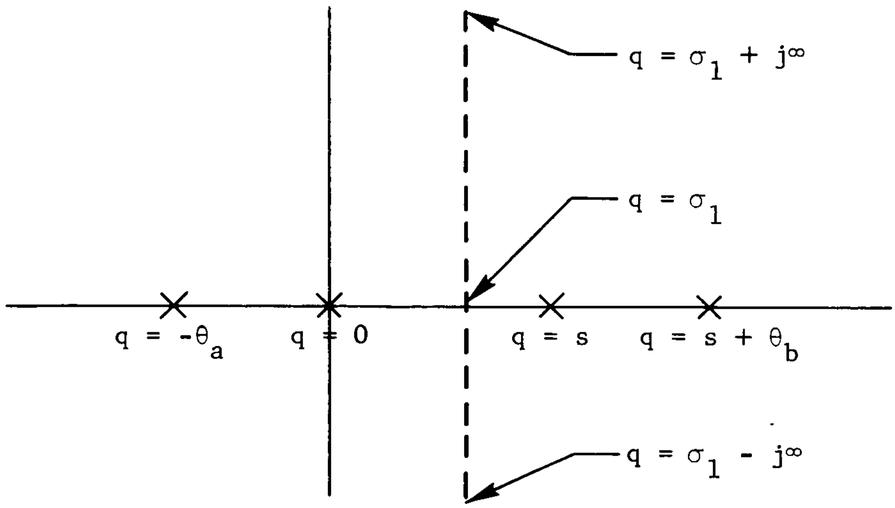
In general,[25][26] σ_1 is computed from $\sigma' < \sigma_1 < \sigma - \sigma''$

where $\sigma' \equiv \mathrm{Re}(s)$ for which $\frac{L_a(s)}{s}$ converges,

$\sigma'' \equiv \mathrm{Re}(s)$ for which $\frac{L_b(s)}{s}$ converges,

and $\sigma \equiv \max\,(\sigma', \sigma'', \sigma' + \sigma'')$.

Looking at the singularities of $\frac{1}{s} L_{12}(s)$ in the q-plane, it is seen that the σ_1-line separates them as discussed above.



By Cauchy's residue theorem the integral can be evaluated by determining the residues of the function at the poles enclosed in the contour containing the σ_1-line. Now the contour can be constructed to the left or to the right as long as the function converges within the contour. For this example the left contour is selected and the residues at $q = 0$ and $q = -\theta_a$ are required. The residue at $q = 0$ is

$$\lim_{q \to 0} \left\{ q \left[\frac{1}{q} \left(\frac{\theta_a}{\theta_a + q} \right) \left(\frac{1}{s - q} \right) \left(\frac{\theta_b}{\theta_b + s - q} \right) \right] \right\} = \frac{1}{s} \left(\frac{\theta_b}{\theta_b + s} \right) .$$

The residue at $q = -\theta_a$ is

$$\lim_{q \to -\theta_a} \left\{ (\theta_a + q) \left[\frac{1}{q} \left(\frac{\theta_a}{\theta_a + q} \right) \left(\frac{1}{s - q} \right) \left(\frac{\theta_b}{\theta_b + s - q} \right) \right] \right\} = \frac{-\theta_b}{(s + \theta_a)(\theta_b + s + \theta_a)} .$$

From the above

$$\frac{1}{s} L_{12}(s) = \frac{1}{s} \left(\frac{\theta_b}{\theta_b + s} \right) - \frac{\theta_b}{(s + \theta_a)(s + \theta_a + \theta_b)} ,$$

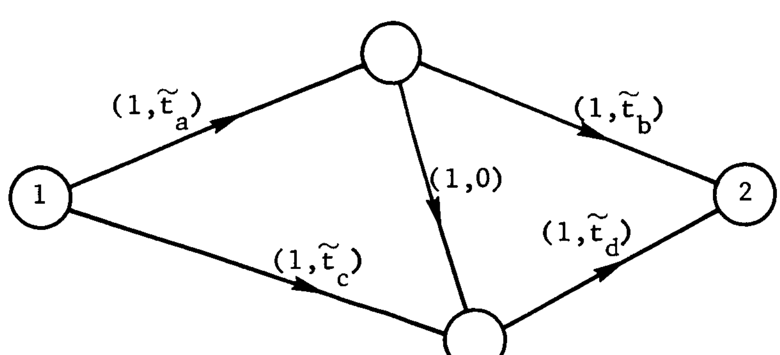
which can be shown to be the Laplace transform of the distribution function of the maximum of two random variables, each having exponential distributions with parameters θ_a and θ_b. As an extension of the above, the time element $\tilde{t}_c$ is added to both the branches, i.e., it is desired to determine the max $(\tilde{t}_a + \tilde{t}_c; \tilde{t}_b + \tilde{t}_c)$. Since max $(\tilde{t}_a + \tilde{t}_c; \tilde{t}_b + \tilde{t}_c)$ equals $\tilde{t}_c$ + max $(\tilde{t}_a; \tilde{t}_b)$, we have $L_{13}(s) = L_{12}(s)\ L_c(s)$ where $L_c(s)$ is the Laplace transform of $\tilde{t}_c$. Thus if

$$L_c(s) = \frac{\theta_c}{\theta_c + s}$$

then $$L_{13}(s) = \left(\frac{\theta_b}{\theta_b + s}\right)\left(\frac{\theta_c}{\theta_c + s}\right) - \frac{s\ \theta_b\ \theta_c}{(s + \theta_a)(s + \theta_a + \theta_b)(s + \theta_c)} .$$

This example demonstrates the usefulness of working entirely in the s-plane. Of course it would be possible to treat the random variables directly; however, the premise is that the networks contain EXCLUSIVE-OR nodes that can provide the L(s) function through the use of the topological equation. The main drawback to this approach is the difficulty of the complex convolution. This difficulty is even more severe when dealing with discrete random variables because the Laplace transforms of many discrete functions have an infinite number of singularities.

Before leaving this proposed method of approach, the network previously analyzed with a "dummy" crossover branch will be studied. The network is redrawn for convenience and the equation for the distribution functions rewritten:



$$\text{Prob}\,(\tilde{t}_{12} \le t) = \text{Prob}\,(\tilde{t}_b \le t - \tilde{t}_a;\ \tilde{t}_d \le t - \tilde{t}_a)[1 - \text{Prob}\,(\tilde{t}_a \le t)]\ \text{Prob}\,(\tilde{t}_c \le t)$$

$$+ \text{Prob}\,(\tilde{t}_b \le t - \tilde{t}_a;\ \tilde{t}_d \le t - \tilde{t}_c)\{1 - [1 - \text{Prob}\,(\tilde{t}_a \le t)][\text{Prob}\,(\tilde{t}_c \le t)]\}\ .$$

Taking the Laplace transform yields

$$\frac{1}{s} L_{12}(s) = \left[\frac{1}{s} \cdot L_a(s) \cdot s \cdot \left(\frac{L_b(s)}{s} \otimes \frac{L_d(s)}{s}\right)\right] \otimes \left[\frac{1}{s} - \frac{L_a(s)}{s}\right] \otimes \left[\frac{L_c(s)}{s}\right]$$
$$+ \left[\frac{L_a(s) L_b(s)}{s}\right] \otimes \left[\frac{L_c(s) L_d(s)}{s}\right] \otimes \left\{\frac{1}{s} - \left[\frac{1}{s} - \frac{L_a(s)}{s}\right] \otimes \frac{L_c(s)}{s}\right\},$$

where $\otimes$ indicates the complex convolution operation. The complexity of this approach can thus be seen. As a possible approximation, the L(s) functions could be put into polynomial form and truncated at the number of moments desired. This is a possibility for future research.

AND Nodes and Inversion Methods. In the foregoing discussion it was seen that the maximum of two random variables or, equivalently, the product of two distribution functions can be calculated by performing a convolution in the complex plane. From previous results the sum of two random variables can be calculated as a product in the transform space. An approach to the analysis of AND nodes is to use transform and inverse transform methods so that only products need be computed and thus the convolution operation would be avoided. Inversion methods that are appropriate are the complex inversion formula for Laplace transforms, Gram-Charlier Series expansion, and using the moments to fit a Pearson-type curve. The specific inversion method would depend on the problem being solved.

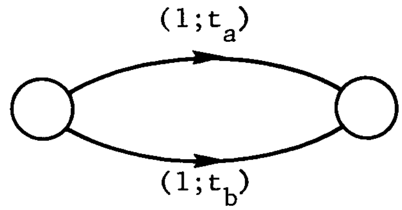
A word of caution needs to be injected here. It is anticipated that the w(s)- (or L(s)) functions will involve many terms. If this were not the case, the analysis could probably be performed entirely in the time domain. Thus the inversion formula may not be computationally tractible. The Gram-Charlier series offers promise in that it approximates the distribution functions by a polynomial whose coefficients are a function of the moments of the distribution. Since the moments are obtainable from the w(s) function, the Gram-Charlier series can be obtained. Given a polynomial representation for a distribution function, J. J. Martin[27] has proposed a method of analysis for networks involving only AND nodes, which involves a computer routine for convolving two polynomials. The problem of dependence of branches has, however, not been programmed.

The above discussion again presents meaty problems for future research. At this time a special case of GERT networks with AND nodes will be examined. The case of interest is when all times on the GERT network are constants and the variability in the network duration is due entirely to path selection.

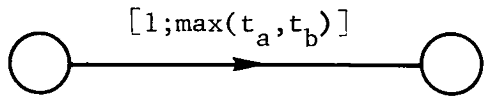
ANALYSIS OF NETWORKS WITH AND NODES AND CONSTANT TIMES

The approach to be pursued here is to develop a method of approach for analyzing a network which can be reduced to all AND nodes and which has branches whose w-functions are of the form $\sum_i p_i e^{st_i}$. The w-functions will be in this form if there are no feedback paths in between AND nodes of the network. Extension to the case where feedback is permitted will then be discussed.

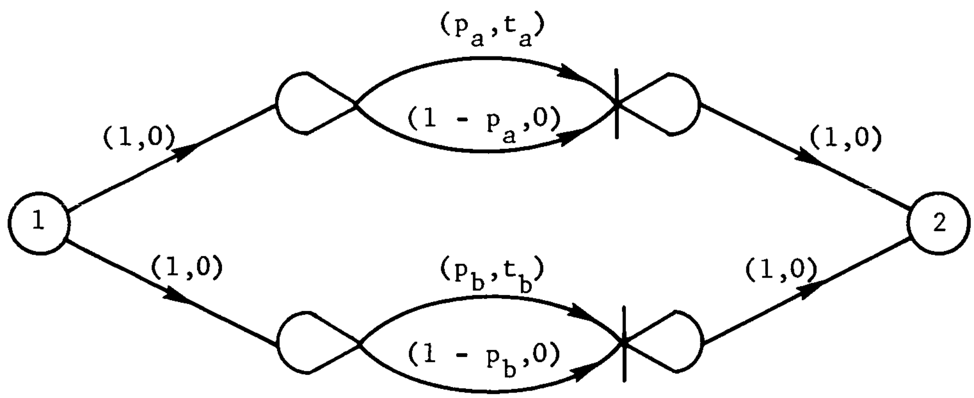
A few examples will demonstrate the approach. It has been shown that the network



is equivalent to



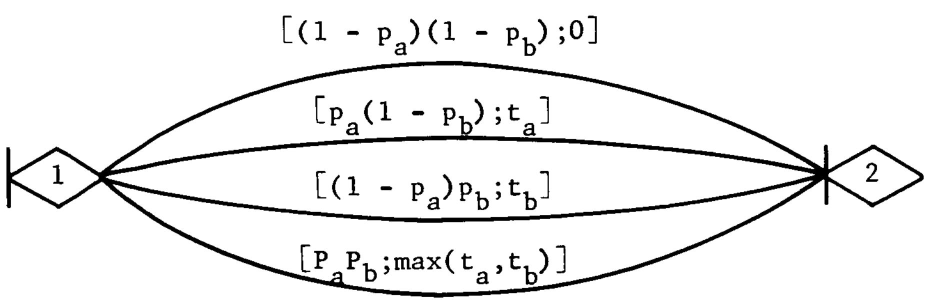
. Consider the network



There are four possible outcomes regarding the length of time to go from 1 to 2. These are:

Outcome	Probability	Time
1	$(1 - P_a)(1 - P_b)$	0
2	$p_a(1 - p_b)$	t_a
3	$(1 - P_a)P_b$	t_b
4	$p_a\ p_b$	max $(t_a;\ t_b)$

The equivalent network in EXCLUSIVE-OR form is

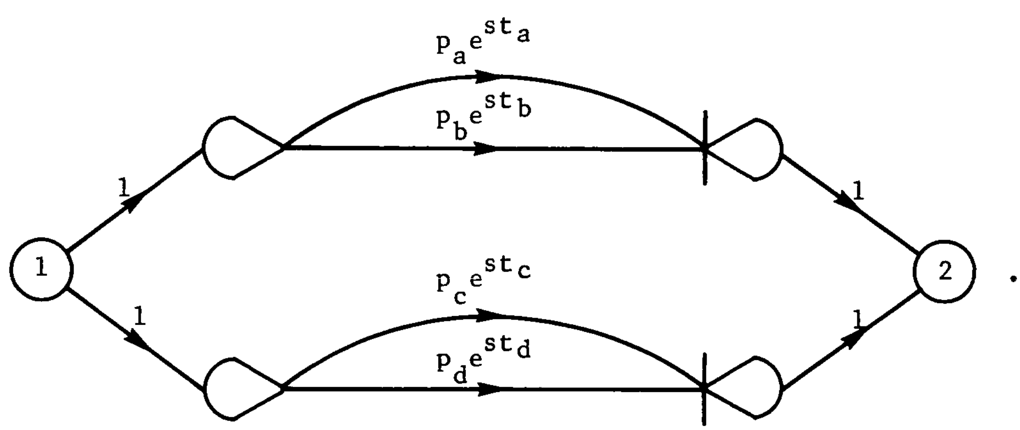


,

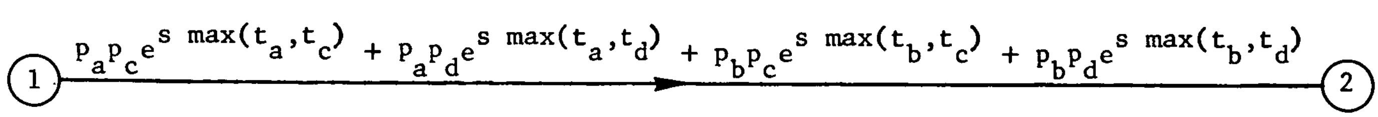
and the moment generating function for this network is

$$M_{12}(s) = (1 - p_a)(1 - p_b) + p_a(1 - p_b)e^{st_a} + (1 - p_a)p_b\, e^{st_b}$$
$$+ p_a p_b e^{s \max(t_a, t_b)} \; .$$

To add further insight into the reduction procedure, another illustration is given below:



where $p_a + p_b = 1$ and $p_c + p_d = 1$. The equivalent network is



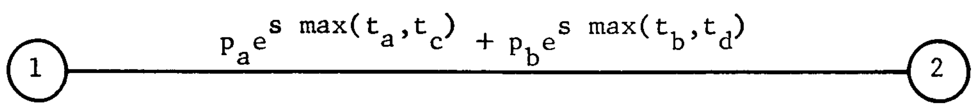
If the probabilities are independent, i.e., $p_{i \cap j} = p_i p_j$ for all i and j. In this case the density function is given by

$f_{12}(t)$	$\tilde{t}_{12}$
$p_a p_c$	$\max\,(t_a, t_c)$
$p_a p_d$	$\max\,(t_a, t_d)$
$p_b p_c$	$\max\,(t_b, t_c)$
$p_b p_d$	$\max\,(t_b, t_d)$

Suppose in the above example that p_c and p_d are not independent of p_a and p_b. For example, the condition could be imposed that if branch a is realized, then branch c must also be realized, and similarly for branches b and d. In this case

$$p_{a\cap c} = p_a = p_c, \; p_{b\cap c} = 0, \; p_{a\cap d} = 0, \text{ and } p_{b\cap d} = p_b = p_d \; .$$

For this case the equivalent network is

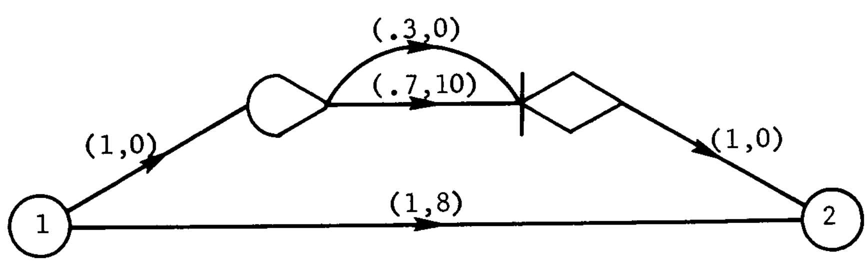


In general, for two branches in parallel between AND nodes which have w-functions of the form

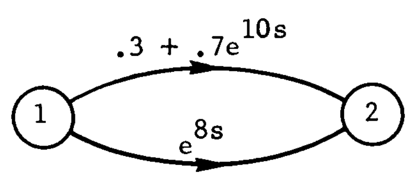
$$\sum_i p_i e^{st_i} \quad \text{and} \quad \sum_j p_j e^{st_j} ,$$

then the w-function of the equivalent network is $\sum_i \sum_j p_{i\cap j} e^{s\max(t_i,t_j)}$.

The problem posed at the beginning of this section can now be solved. The network presented was:

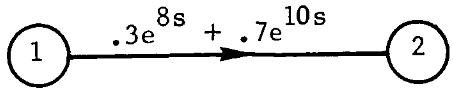


Transforming to M.G.F. yields



,

which, according to the above results in the equivalent network,



For this equivalent network, we have

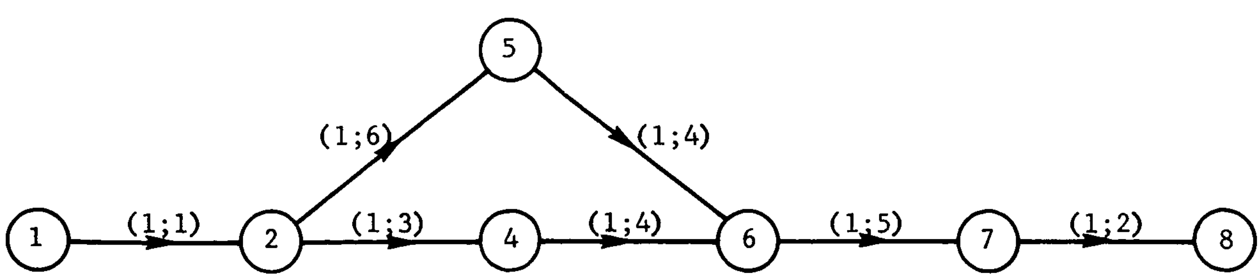
$$\mu_{1E} = \frac{d}{ds}\left[\frac{.3e^{8s} + .7e^{10s}}{1.0}\right]_{s=0} = 2.4 + 7 = 9.4$$

and

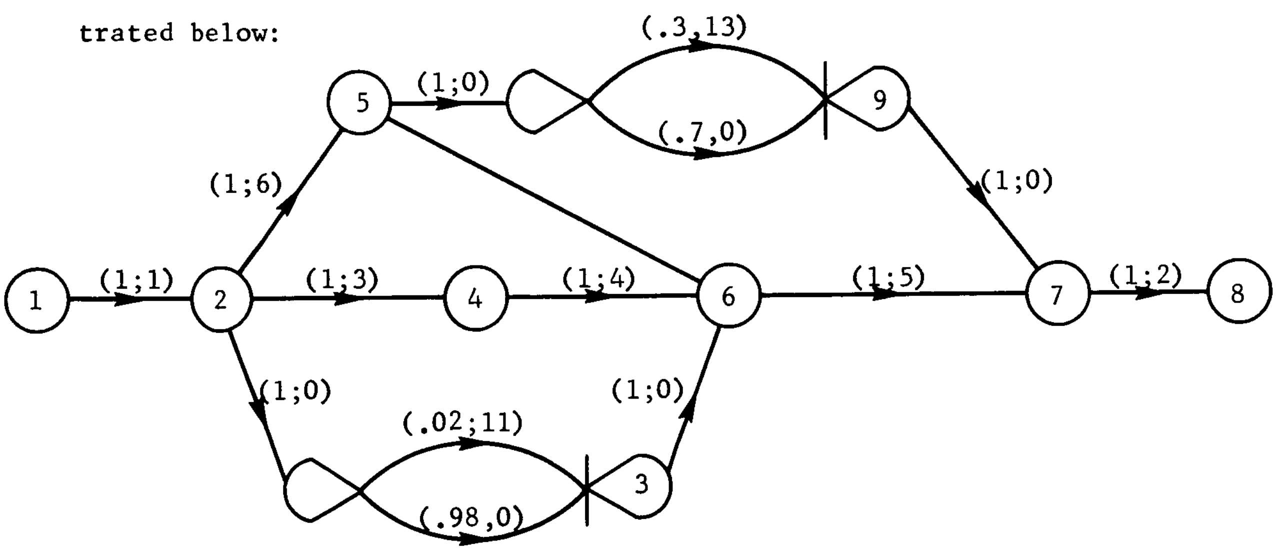
$$\mu_{2E} = \frac{d^2}{ds^2}\left[\frac{.3e^{8s} + .7e^{10s}}{1.0}\right]_{s=0} = 19.2 + 70 = 89.2 \ .$$

A procedure has now been developed for combining parallel branches. The next step is to provide a procedure for reducing combinations of parallel and series branches. The procedure proposed is to alter the network by adding branches that make parts of the network dependent on other paths, but causes all paths to be exclusively of the parallel type or the series type. In this manner, paths will result that can be reduced by the procedures discussed above. The adding of branches will be done in such a manner that there will be no effect on the measures of performance associated with the network. This procedure will be explained by example.

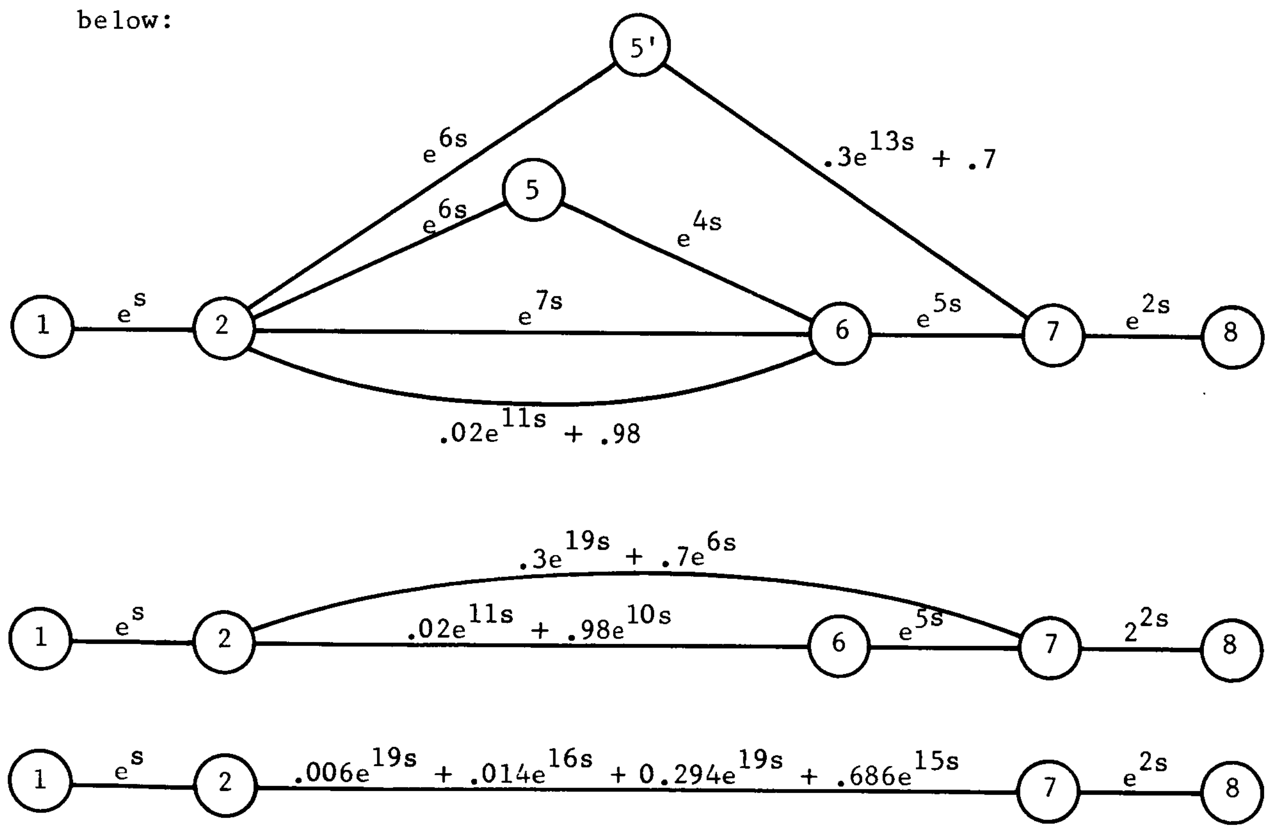
Assume that there is a nominal schedule of activities for a project, as shown below in network form:



From the network it is seen that all activities (branches) must be performed and that each activity requires a constant amount of time. Suppose that two branches are superimposed on this network between nodes 2 and 6 and between 5 and 7. These new branches represent activities that do not have to be performed all the time and, indeed, are only performed say 2 and 30 per cent of the time, respectively. Examples of such activities would be the need of a repair action, a spare part, a demand for a specific service, and the like. According to a previous discussion, two branches must be added for each new activity because, if the new activity is not performed, it must be shown in the network in order to make it complete. This is illustrated below:



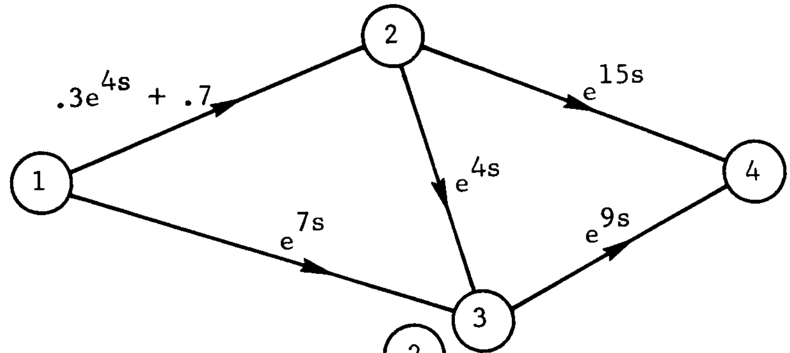
The addition of the branches between nodes 2 and 3 does not cause any difficulty, since they are in parallel with the series combination between nodes 2 and 6. Thus parallel paths exist and reduction would proceed as previously discussed. This is not the situation for the branches added between nodes 5 and 9. To circumvent this interdependence an extra node is added to the network, say 5', and node 2 will be connected to node 5' by a branch with the same characteristics as the branch between nodes 2 and 5. The addition of this branch does not affect the network (nor in this case the distribution of the times to reach any node), since the branch added will take six time units with certainty. For such a case, the activities are independent and the probability of both activities occurring equals the product of probabilities. The reduction procedure can now proceed in the steps shown below:



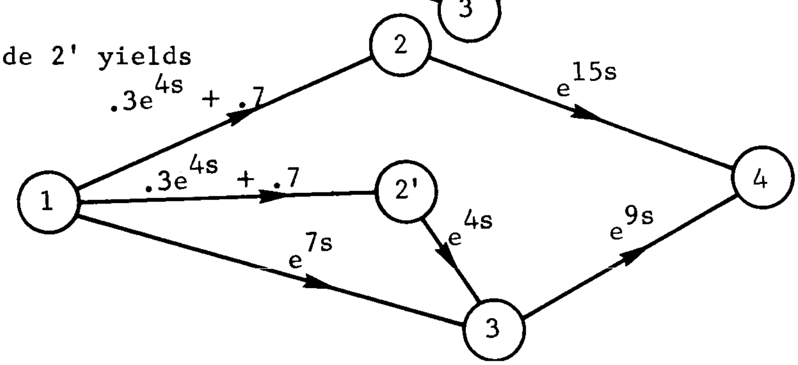
The moments can now be obtained as discussed in the section on EXCLUSIVE-OR logic elements. For this example the M.G.F. is

$$M_E(s) = 0.30e^{22s} + 0.014e^{19s} + 0.686e^{18s} \; .$$

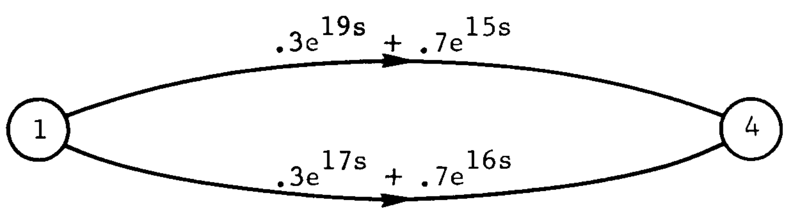
In the above example the dependence between the branches between nodes 2 and 5 and nodes 2 and 5' did not complicate the analysis, since the probability of taking both branches was one. An example in which this is not the case is given below.



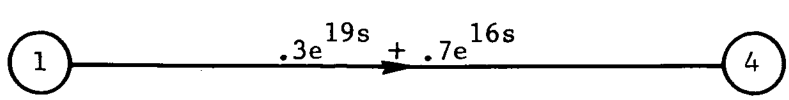
Adding node 2' yields



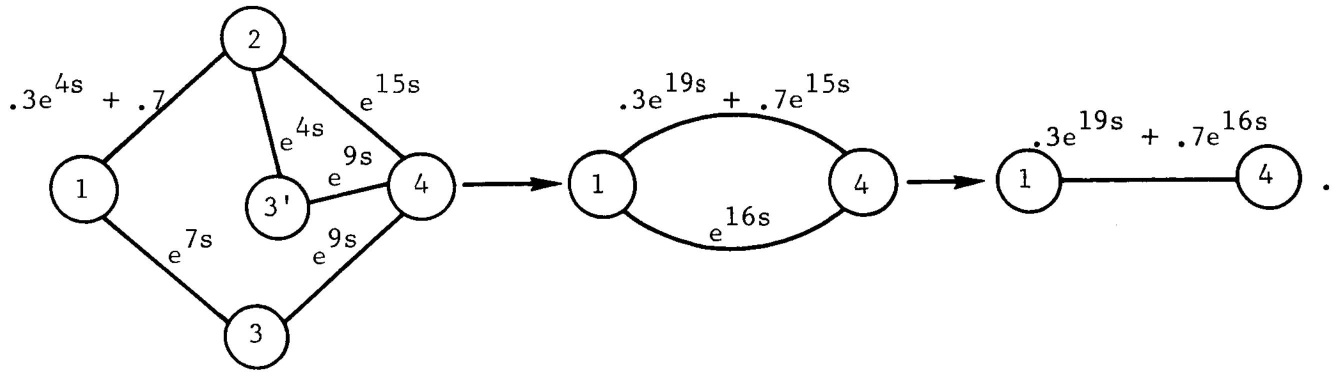
The network now consists of paths which can be reduced by using the parallel and series reduction rules to obtain



From this network it is seen that the upper branch takes 19 time units 30 per cent of the time and 15 time units 70 per cent of the time. Since the only probabilistic branch was the one which was added, it is observed that the lower branch will take 17 time units 30 per cent of the time, but this will occur only when the upper branch takes 19 time units. Thus, according to the previous discussion, the equivalent network is



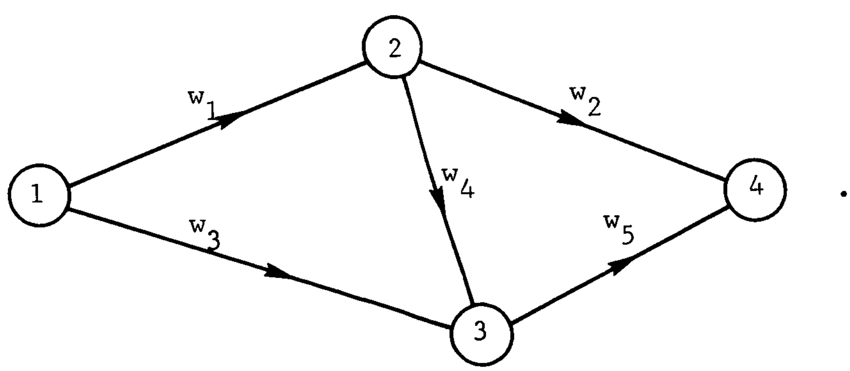
To see that this is the correct result, the network is reanalyzed in the following manner:



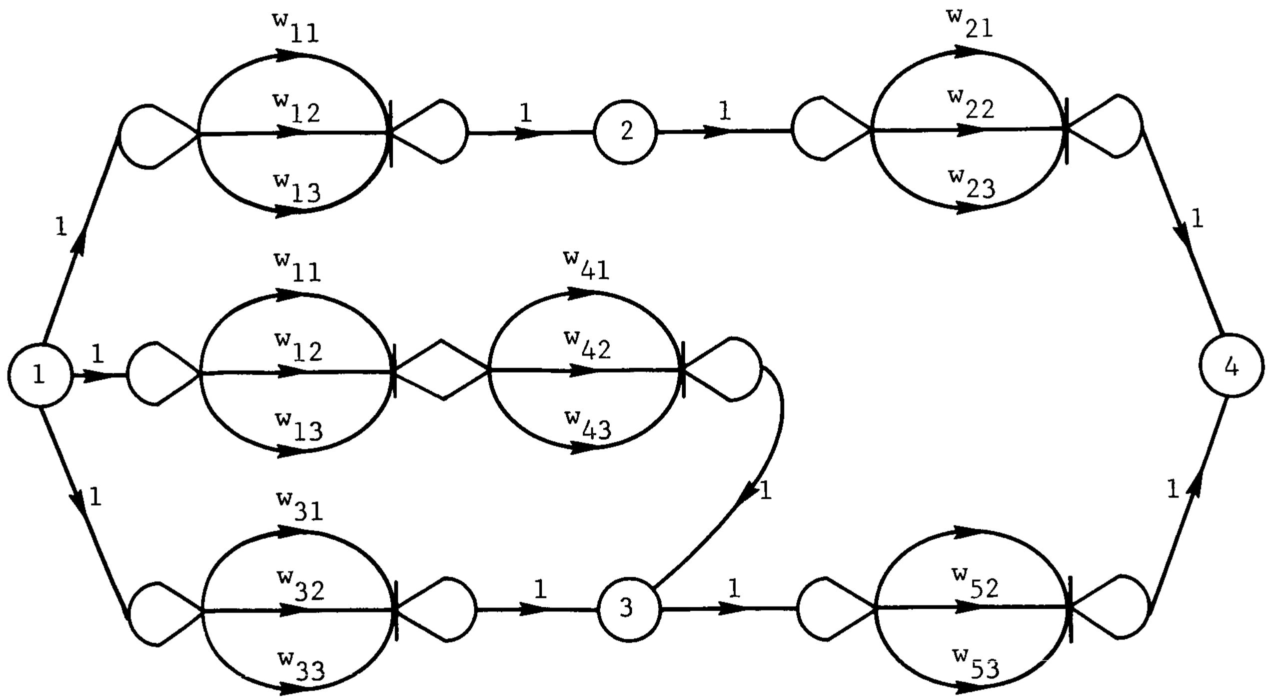
This example demonstrates that if the added branch occurs with certainty, the problem of dependence can be overcome. A situation where this cannot be accomplished is presented in Example 15.

Example 15. A Simple PERT-Type Network with Probabilistic Branches

To illustrate the complexity of the AND nodes, a "simple" PERT-type network will be evaluated. The network consists of four nodes (milestones) and five branches (activities) as shown below:



Associated with each activity are three time values with the probability that these time values are realized. Including each of these possible branches on the network yields



where activity 1 is shown twice to make parallel paths on the graph.

The M.G.F. of the time to go from node 1 to node 4 is

$$M_E(s) = \sum_{U=1}^{9} \sum_{L=1}^{81} p_{U \cap L} e^{s \max(t_U; t_L)} ,$$

where

$$p_U = p_{1i} \, p_{2j} \qquad i, j = 1, 2, 3 \quad \text{and } U = i + 3 (j - 1)$$

$$t_U = t_{1i} + t_{2j} ,$$

$$p_L = p_{1i} \, p_{4j} \, p_{3k} \, p_{5m} \qquad i, j, k, m = 1, 2, 3 \quad \text{and} \quad L = i + 3 (j - 1)$$

$$3(k - 1) + 3 (m - 1)$$

$$t_L = \max (t_{1i} + t_{4j}; t_{3k}) + t_{5m} ,$$

$$p_{1a \cap 1b} = \begin{cases} p_{1a} & \text{if } a = b \\ 0 & \text{otherwise, and all other probabilities are independent.} \end{cases}$$

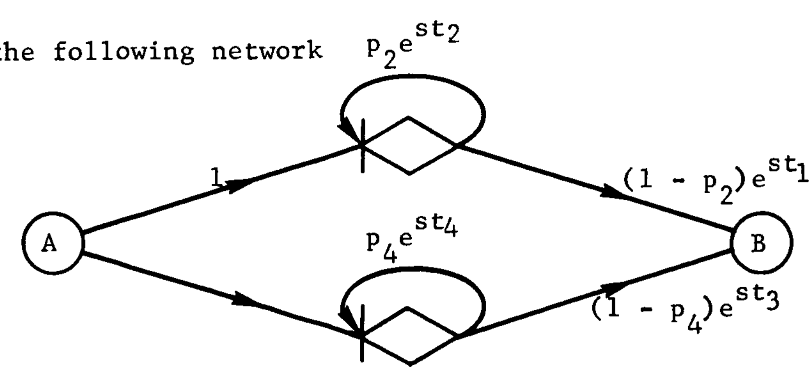
For this simple problem there are 729 possible realizations of the network, and hence the probability law associated with the time to realize node 4 will have 729 possible values. (The above process is equivalent to complete enumeration but presents a consistent mechanical method.) Only with digital computation procedures will more complex networks of this type be computationally tractable.

This example demonstrates that GERT eliminates the need for making assumptions about the distribution of the time associated with a branch and the inherent error involved. Also, a procedure for obtaining the exact M.G.F. is provided. If for large networks the computations are excessive, approximation techniques can be employed to remove branches which are not critical. If only the expected value is of interest, the approximation techniques given in Refs. 3 and 4 can be used.

The above descriptions have been presented to describe the problem and to illustrate clearly the difficulties to be anticipated. The general problem for constant times will be presented below and the solution for one special case given.

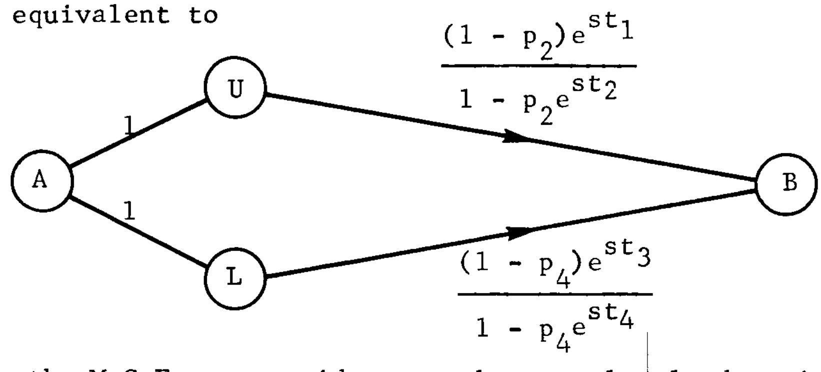
General Analysis of Networks with AND Nodes and Constant Times

The networks discussed in the above paragraphs did not include feedback branches. The introduction of w-functions that contain feedback elements further complicates the analysis of AND nodes. Consider the following network



,

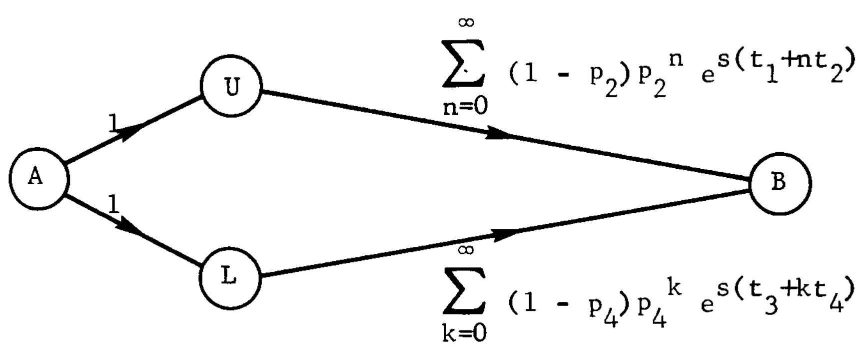
which is equivalent to



Since for the M.G.F. we consider s to be a real-valued variable, we can expand $\frac{1}{1 - pe^{st}}$ in a power series and obtain

$$\frac{1}{1 - pe^{st}} = \sum_{n=0}^{\infty} (pe^{st})^n \ .$$

Inserting this relationship on the above network and combining terms yields



This network demonstrates that for the analysis of AND nodes it is necessary to combine infinite series according to the conditions of the AND logic operation. For the above network the equivalent time to realize node B will be $\tilde{t}_{UB}$ if $\tilde{t}_{UB} \geq \tilde{t}_{LB}$ and $\tilde{t}_{LB}$ if $\tilde{t}_{UB} < \tilde{t}_{LB}$.

Thus $$\text{Prob}\ (\tilde{t}_{AB} = t) = \text{Prob}\ (\tilde{t}_{UB} = t)\ \text{Prob}\ (\tilde{t}_{UB} \geq \tilde{t}_{LB})$$

$$+ \ \text{Prob}\ (\tilde{t}_{LB} = t)\ \text{Prob}\ (\tilde{t}_{UB} < \tilde{t}_{LB}) \ .$$

For the network with feedback paths, but only constant times, the possible values of t are discrete and limited to $t_1 + nt_2$ and $t_3 + kt_4$ for all integer n and k. Consider the case $t = t_1 + nt_2$. It can be shown that

$$\text{Prob } (\tilde{t}_{UB} = t_1 + nt_2) = (1 - p_2)p_2^{\,n}.$$

Now for $\tilde{t}_{UB} \geq \tilde{t}_{LB}$, it is necessary for $t_1 + nt_2 \geq t_3 + kt_4$, or for $k \leq \dfrac{t_1 + nt_2 - t_3}{t_4}$. But k must be an integer (a loop must be traversed an integer number of times), hence $k \leq \left[\dfrac{t_1 + nt_2 - t_3}{t_4}\right]$ where $[x] = 2$ if $2 \leq x < 3$. With this information it is seen that

$$\begin{aligned}
\text{Prob } (\tilde{t}_{AB} = t_1 + nt_2) &= (1 - p_2)\, p_2^{\,n} \sum_{k=0}^{\left[\frac{t_1 + nt_2 - t_3}{t_4}\right]} (1 - p_4)p_4^{\,k} \\
&= (1 - p_2)p_2^{\,n}\left(1 - p_4^{\left[\frac{t_1 + nt_2 - t_3}{t_4}\right] + 1}\right) \\
&= (1 - p_2)p_2^{\,n}\left(1 - p_4^{\left[\frac{t_1 + nt_2 - t_3 + t_4}{t_4}\right]}\right)
\end{aligned}$$

for n = 0, 1, 2, ...

Letting $t = t_3 + kt_4$, we have

$$\text{Prob}\ (\tilde{t}_{AB} = t_3 + kt_4) = (1 - p_4)p_4^{\,k} \sum_{n=0}^{\left[\left[\frac{t_3 + kt_4 - t_1}{t_a}\right]\right]} (1 - p_2)p_2^{\,n}$$

$$= (1 - p_4)p_4^{\,k} \left(1 - p_2^{\left[\left[\frac{t_3 + kt_4 - t_1 \pm t_2}{t_2}\right]\right]}\right)$$

for $k = 0, 1, 2, \ldots$, and where $[[\quad]]$ is a rounding operation such that $[[\ x\ ]] = 2$ for $2 < x \le 3$. Combining these terms and transforming into the w notations results in the following network:

$$\text{(A)} \quad \sum_{n=0}^{\infty} (1 - p_2)p_2^{\,n}\, e^{s(t_1 + nt_2)} \left(1 - p_4^{\left[\frac{t_1 + nt_2 - t_3 + t_4}{t_4}\right]}\right) +$$

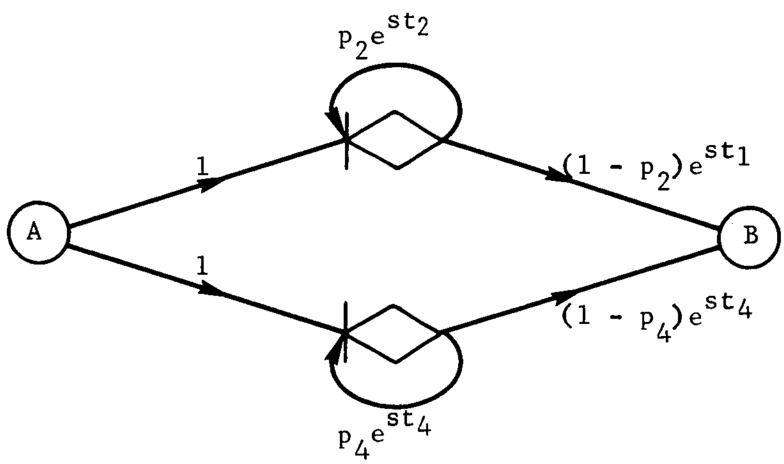
$$\sum_{k=o}^{\infty} (1 - p_4)\, p_4^{\,k}\, e^{s(t_2 + kt_4)} \left(1 = p_2^{\left[\left[\frac{t_3 + kt_4 - t_1 + t_2}{t_2}\right]\right]}\right) \quad \text{(B)}$$

Thus for a simple network with only two paths in parallel with each path having only one feedback loop, the resulting w-function is not in a simple form.

To illustrate the form of the density functions, three numerical examples are given in Table 12.

Table 12

DENSITY FUNCTIONS ASSOCIATED WITH THE NETWORK



p_2	.8		.3		.3	
p_4	.7		.2		.5	
t_1	1		3		1	
t_2	2		7		4	
t_3	3		2		2	
t_4	4		8		6	
	t_{AB}	p_{AB}	t_{AB}	p_{AB}	t_{AB}	p_{AB}
	3	.1080	3	.5600	2	.6300
	5	.0384	10	.3136	5	.0630
	7	.1547	17	.0605	8	.2079
	9	.0418	18	.0311	9	.0082
	11	.1419	24	.0187	13	.0008
	13	.0344	26	.0063	14	.0630
	15	.1132	31	.0057	17	.0001
	17	.0255	34	.0013	20	.0189
	19	.0847	38	.0017	21	.0000
	21	.0179	42	.0003		
$\sum p_{ab}$		.7605		.9992		.9919

Two observations concerning this illustration are worth noting: (1) the resulting terms of the w-function are in the form pe^{st}; and (2) the number of terms of the form pe^{st} to account for most of the density function is small if the probabilities of taking the feedback paths are small.

A network with two branches in parallel leading into an AND node, where the branches have w-functions of arbitrary complexity, can be analyzed in a manner similar to the above. From Mason's form of the topological equation, the denominator of an equivalent w-function is in the form

$$1 - \sum w_{I_1} + \sum w_{I_2} w_{I_3} - \sum w_{I_4} w_{I_5} w_{I_6} + \ldots,$$

where the I_j are index sets such that $I_j \subset I_1$ for $j > 1$. To obtain all the terms of the equivalent w_E function, the reciprocal of this function must be expanded.

Summary of AND Node Analysis

No general method of analysis for AND modes has been developed. Approaches to the problem have been discussed in this appendix and the limitations of the approaches presented. Possible approximation techniques were referred to and simplifying assumptions were considered.

Basically the methods proposed are:

1. Solve the complex convolution of two Laplace transforms;
2. Invert to the domain in which only multiplication need be performed;
3. Reduce the network to all AND nodes, using the w- or L-function, then transforming to the time domain and employing a suitable analysis in the time domain, such as is currently done in PERT or by using the algorithm suggested by J. J. Martin;

4. Approximate the w- or L-function by a polynomial in s and derive a method for combining the resulting polynomials according to the AND logical operation; and

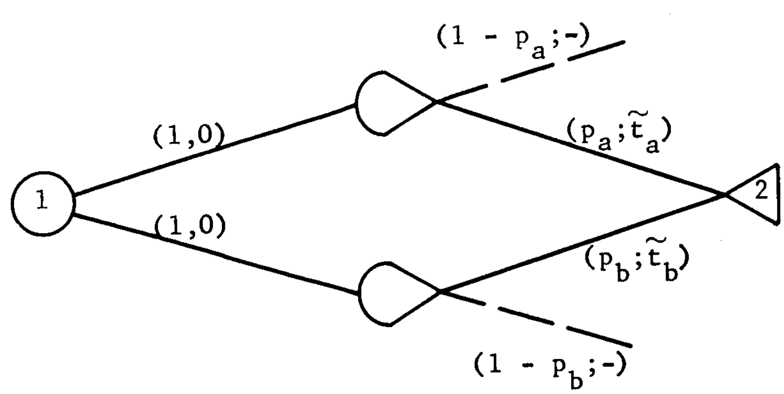
5. Use a brute force approach to the solution of networks that only have constant time parameters.

For the present it appears that the specific problem will determine the appropriate procedure to follow. Analysis of AND nodes represents a fertile area for future research.

THE INCLUSIVE-OR LOGIC ELEMENT

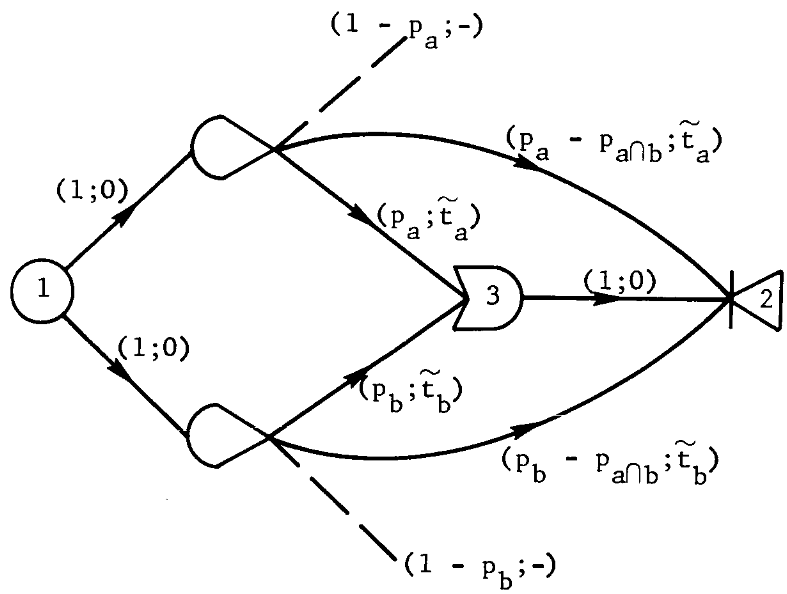
Only a brief discussion of the INCLUSIVE-OR node will be presented here, since it has many characteristics that are similar to the AND node. An INCLUSIVE-OR node is one in which the realization of any of the branches incident to the node causes the node to be realized. The INCLUSIVE-OR node differs from the EXCLUSIVE-OR node in that more than one of the incident branches can be realized.

As discussed in Sec. I for the simple network



,

there are three possible ways that node 2 can be realized: (1) a is realized but not b; (2) b is realized but not a; and (3) both a and b are realized. The simple network can be redrawn as



,

where the node symbol $\sum$ indicates a MINIMUM node. The MINIMUM node will be realized only if all branches incident to the node are realized. The time at which the MINIMUM node is realized is the minimum of the times at which the branches leading to the node are realized. Thus for the above example, the probability of realizing node 3 is $p_{a\cap b}$, and $\tilde{t}_{13} = \min(\tilde{t}_a, \tilde{t}_b)$. The distribution function of the time to traverse from node 1 to node 3, given that node 3 is realized, is

$$\begin{aligned}\text{Prob}(\tilde{t}_{13} \le t) &= \text{Prob}(\tilde{t}_a \ge t;\ \tilde{t}_b \ge t) \\ &= \text{Prob}(\tilde{t}_a \ge t)\ \text{Prob}(\tilde{t}_b \ge t)\end{aligned}$$

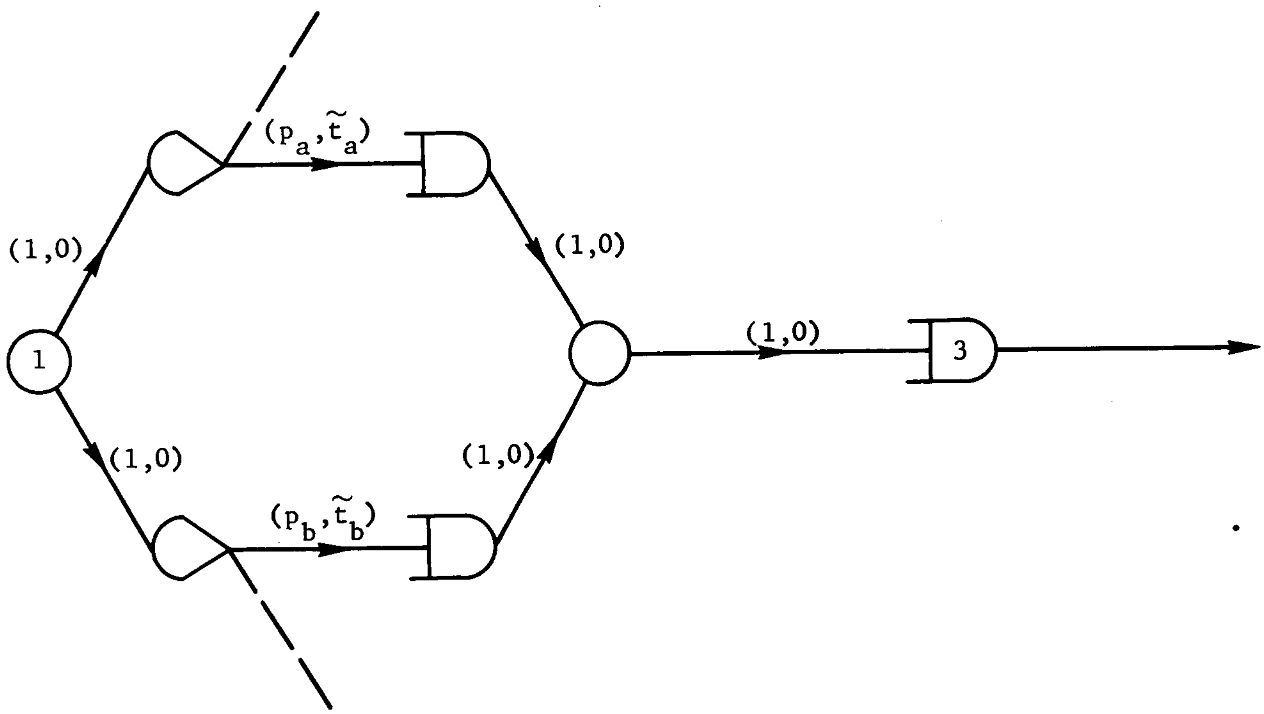
if t_a and t_b are independent. Thus

$$F_{13}(t) = (1 - F_a(t))\ (1 - F_b(t)).$$

This is seen to be analogous to the AND node. This analogy leads to the definition of another logical operation, the INVERTOR, whose input side is symbolized by I. The INVERTOR node can have only one input branch. The role of the INVERTOR is to obtain the additive inverse for the time parameter. The application of the INVERTOR node to the above network requires the following observation.

$$\tilde{t}_{13} = \min(\tilde{t}_a, \tilde{t}_b) = -\max(-\tilde{t}_a, -\tilde{t}_b).$$

Thus, the network from node 1 to node 3 could be drawn as



The use of the INVERTOR node for this example is awkward. The analysis of the INVERTOR node is not amenable to the transform methods previously discussed, since it introduces the concept of negative time. It is presented here solely as a theoretical concept.

In summary, other logical operations will enhance the application of networks to analysis problems; however, the analysis procedure currently must be performed on a specific network basis.

Appendix C

STOCHASTIC NETWORKS WITH MULTIPLICATIVE PARAMETERS

In the development of GERT attention has been restricted to additive parameters. This emphasis on additive parameters, such as time, is justified on the basis of the potential applications. In this appendix attention will be directed to multiplicative parameters, where, for two branches in series, the parameter of the equivalent branch is the product of the parameters of the individual branches. The symbol x will be used to denote the second parameter when the multiplicative property is being discussed.

Two logic nodes will be discussed here: (1) the EXCLUSIVE-OR node; and (2) the logic node employed in flowgraph theory. In both cases methods for including random variables will be developed.

THE EXCLUSIVE-OR NODE

The equivalent networks for series, parallel, and self-loop networks, when the x parameter is a constant, are given in Fig. 11. The expressions shown in Fig. 11 are developed in the same manner as was discussed for the additive parameter. For two branches in series both the p and x values are multiplied. For the parallel network, either branch can be taken (but not both), and the same result as for the additive parameter branches is derived. In the self-loop network, the expected value of x is obtained by enumeration.

Based on the previous developments, it would be possible to determine equivalent networks for complex networks if p and x can be combined into a single quantity. The appropriate transform for a

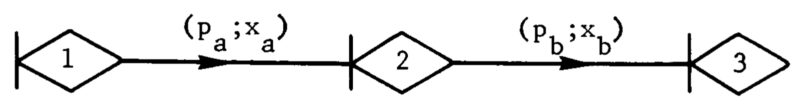
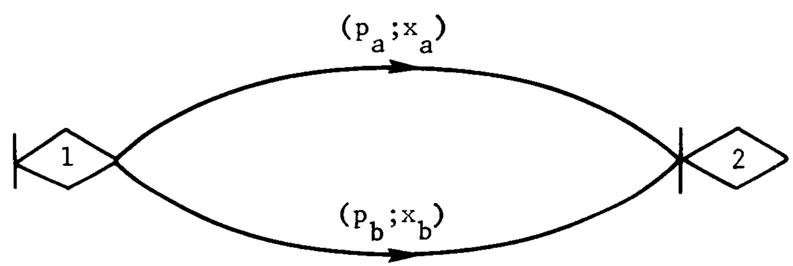
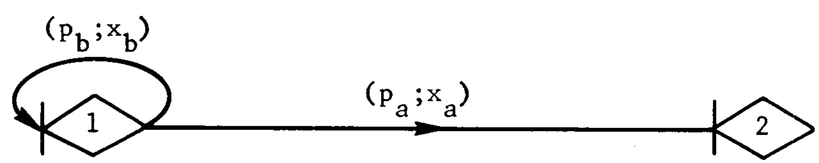
Network type	Representation with constant times	Equivalent probability	Equivalent expected value of x
Series		$p_a\ p_b$	$x_a\ x_b$
Parallel		$p_a + p_b$	$\frac{p_a x_a + p_b x_b}{p_a + p_b}$
Self-loop		$\frac{p_a}{1 - p_b}$	$\frac{(1 - p_b)x_a}{1 - p_b x_b}$ if $\lvert p_b x_b \rvert < 1$ ∞ otherwise

Fig. 11--Equivalent network results for multiplicative parameters

multiplicative parameter is the Mellin transform,[(28)(29)] which is defined by

$$G(s) = \int_o^\infty x^{s-1} f(x)\,dx \quad .$$

The following simplified analysis shows that the Mellin transform is appropriate for products of random variables: let $u = xy$, then

$$\ell n\ u = \ell n\ x + \ell n\ y \ .$$

Consider the M.G.F. of $\ell n\ x$, $M_{\ell n\ x}(s)$,

$$M_{\ell n\ x}(s) = E\{e^{s\ \ell n\ x}\} = E\{e^{\ell n\ x^s}\} = E\{x^s\} \quad .$$

Now from the fact that the M.G.F. of the sum of two random variables is the product of the M.G.F. of each random variable, we have

$$M_{\ell n\ u}(s) = M_{\ell n\ x}(s)\ M_{\ell n\ y}(s)$$

or

$$E\{u^s\} = E\{x^s\}E\{y^s\}$$

by the above. It is easily shown that $E\{x^s\}$ is related to $E\{x^{s-1}\}$, which is the Mellin transform, $G(s)$. Combining p with $G(s)$ for the same reasons as in the additive case, we have $V(s) = p\ G(s)$. The quantity $V(s)$ completely characterizes stochastic networks with EXCLUSIVE-OR nodes and two multiplicative parameters.

Before demonstrating the use of the transformation, several characteristics of $G(s)$ and $V(s)$ should be noted. In $G(s)$, values of x are restricted to be positive. For some functions this restriction can be obviated.[23] If f(x) is a probability density function, then $G(s) = E\{x^{s-1}\}$ so that

$$E\{x\} = G(s)\big|_{s=2}$$

and

$$E\{x^n\} = G(s)\big|_{s=n+1} \quad .$$

For $x = x_o$ then $G(s) = x_o^{s-1}$ since $\int_o^\infty f(x_o)dx = 1$.

Now V(s) characterizes both the probability that a node will be realized and the Mellin transform of the x parameter associated with the realization of the node. In particular $V_E(1)$ is the probability of realizing the node of interest, p_E. Thus, $G_E(s) = \frac{V_E(s)}{V_E(1)} = \frac{V_E(s)}{p_E}$, and $G_E(n+1)$ is the n^{th} moment of x, given f(x) is a probability distribution function.

The above results can now be applied to specific stochastic networks. In Fig. 12, the use of the V-transform is illustrated for constant x, and the topological equation is used to obtain $V_E(s)$. From $G_E(s)$ in Fig. 12 it is seen that the results regarding expected values given in Fig. 11 are obtained when s = 2. Higher moments are easily obtained.

Consider now that x is distributed according to a negative exponential distribution with a mean $1/\theta$, i.e., $f(x) = \theta e^{-\theta x}$. For this case it can be shown that

$$G(s) = \int_o^\infty x^{s-1}\theta e^{-\theta(x)}dx = \theta^{(1-s)}\Gamma(s)$$

Note that

$$E\{x\} = G(2) = \theta^{-1}$$

and

$$E\{x^2\} = G(3) = 2\theta^{-2} \qquad \text{as expected.}$$

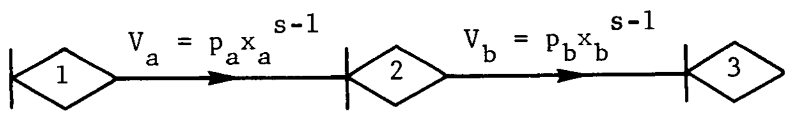
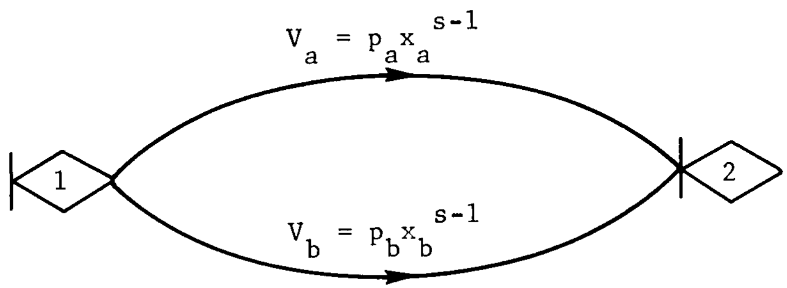
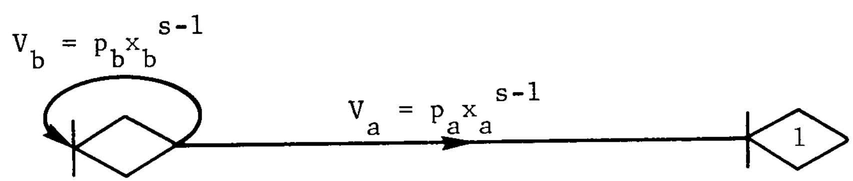
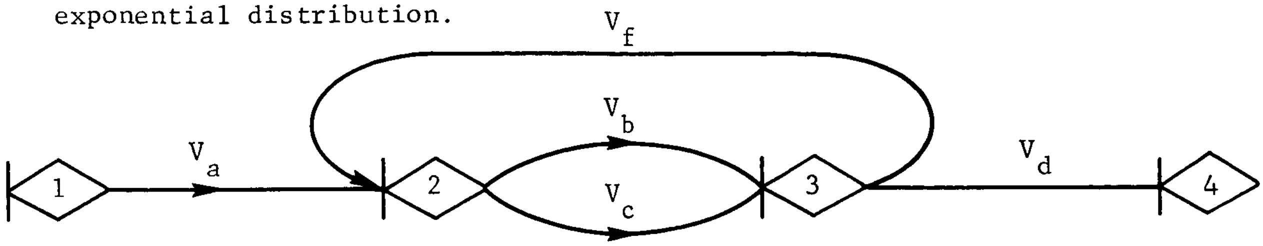
Network Type	Graphical Representation	Equivalent Function V_E	Equivalent Mellin Transform $G_E(s)$
Series		$V_a V_b$	$x_a^{s-1} x_b^{s-1}$
Parallel		$V_a + V_b$	$\dfrac{p_a x_a^{s-1} + p_b x_b^{s-1}}{p_a + p_b}$
Self-loop		$\dfrac{V_a}{1 - V_b}$	$\dfrac{(1 - p_b) x_a^{s-1}}{1 - p_b x_b^{s-1}}$

Fig. 12--Network reduction using topology equation for multiplicative parameter branches

On the following network it will be assumed that all branches have parameters that are random variables described by the negative exponential distribution.



$$V_E = \frac{V_aV_bV_d + V_aV_cV_d}{1 - V_bV_f - V_cV_f} = \frac{V_aV_d(V_b + V_c)}{1 - V_f(V_b + V_c)} .$$

It can be seen from the network that $V_E(1) = 1$, and hence

$$G_E(s) = \frac{p_ap_d[\theta_a^{1-s}\Gamma(s)][\theta_d^{1-s}\Gamma(s)]\ [p_b\theta_b^{1-s}\Gamma(s) + p_c\theta_c^{1-s}\Gamma(s)]}{1 - p_f\theta_f^{1-s}\Gamma(s)\ [p_b\theta_b^{1-s}\Gamma(s) + p_c\theta_c^{1-s}\Gamma(s)]}$$

$$G_E(s) = \frac{p_ap_d[\Gamma(s)]^3[\theta_a^{1-s}\theta_d^{1-s}][p_b\theta_b^{1-s} + p_c\theta_c^{1-s}]}{1 - p_f\theta_f^{1-s}[\Gamma(s)]^2[p_b\theta_b^{1-s} + p_c\theta_c^{1-s}]}$$

where $p_a = 1,$

$p_b = 1-p_c$

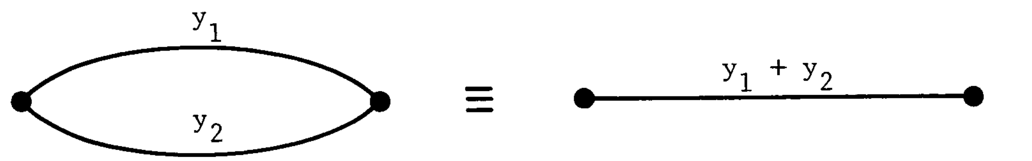
$p_f = 1-p_d.$

From the above equation for $G_E(s)$, all the moments associated with x from node 1 to node 4 can be calculated.

FLOWGRAPHS WITH RANDOM VARIABLES

In Sec. III the computational equivalence between flowgraphs and GERT was discussed. There is, however, a conceptual difference in the logical node used in flowgraphs and in stochastic networks. This difference will be explained in terms of two parallel branches.

For two branches in parallel in a flowgraph, we have



In a flowgraph both branches are employed (realized) and the values are added. There is no question as to realization of the second node. There is no similar concept in stochastic networks, although computationally the probabilities in EXCLUSIVE-OR nodes are computed identically. Because of this difference, the results obtained for multiplicative parameters are not applicable for flowgraphs with random variables associated with the branches.

For flowgraphs with random variables the following scheme is proposed. First, develop the equivalent network in symbolic form using the topological equation. This will yield the equivalent transmittance as a function of the random variables. (Care must be taken when reducing the resulting expression, since the value zero may be included within the range of the random variable.) Second, use the Mellin transform and the M.G.F. to obtain the transform for products and sums of the random variables respectively. This is equivalent to partitioning or segmenting the equivalent network equation into independent parts. Third, if the next operation involves a product (quotient), convert the M.G.F. into Mellin transform form or, if the next operation is a sum, convert the Mellin transform into a Fourier transform.* Continue in this manner until all random variables have

*Mellin transforms are virtually two-sided Laplace transforms and may be expressed either as exponential Fourier transforms in the complex domain, or as combinations of Laplace transforms. (See Ref. 28, p. 305).

been combined. The resulting equation will permit the moments of the equivalent parameter for the entire network to be computed.

REFERENCES

1. MacCrimmon, K.R., and C.A. Ryavec, "An Analytical Study of the PERT Assumptions," Op. Res., Vol. 12, No. 1, 1964, pp. 16-38.

2. Clark, C.E., "The Greatest of a Finite Set of Random Variables," Op. Res., Vol. 9, No., 2, 1961, pp. 145-162.

3. Fullkerson, D.R., "Expected Critical Path Lengths in PERT Networks," Op. Res., Vol. 10, No. 6, 1962, pp. 808-817.

4. Clingen, C.T., "A Modification of Fulkerson's PERT Algorithm," Op. Res., Vol. 12, No. 4, 1964, pp. 629-632.

5. Eisner, H., "A Generalized Network Approach to the Planning and Scheduling of a Research Program," Op. Res., Vol. 10, No. 1, 1962, pp. 115-125.

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

7. Lorens, C.S., Flowgraphs, McGraw-Hill Book Company, New York, 1964.

8. Happ, W.W., "The Application of Flow Graphs to the Solution of Reliability Problems," in Physics of Failure in Electronics, Vol. II, Spartan Books, Inc., 1964.

9. Kim, W.H. and R. T. Chien, Topological Analysis and Synthesis of Communication Networks, Columbia University Press, New York, 1962.

10. Huggins, W.H., "Signal Flow Graphs," Proc. IRE, Vol. 9, No. 9, 1957, pp. 74-86.

11. Howard, R.A., Dynamic Programming and Markov Processes, Technology Press, Massachusetts Institute of Technology, Cambridge, Mass., and John Wiley and Sons, Inc., London, 1960.

12. ____________, "Systems analysis of Semi-Markov Processes," IEEE Transactions on Military Electronics, April 1964, pp. 114-124.

13. Kim, W. H. and R. T. Chien, Topological Analysis and Synthesis of Communication Networks, Columbia University Press, New York, 1962, p. 105.

14. Teicher, H., "Identifiability of Finite Mixtures, Ann. Math. Stat., Vol. 34, 1963, pp. 1265-1269.

15. Rider, P. R., "The Method of Moments Applied to a Mixture of Two Exponential Distributions," Ann. Math. Stat., Vol. 32, 1961, pp. 143-147.

16. Pyke, R., "Markov Renewal Processes: Definitions and Preliminary Properties," Annals of Math. Stat., Vol. 32, 1961, pp. 1231-1242.

17. Whitehouse, G. E., "Extensions, New Developments, and Applications of GERT," unpublished Ph.D. dissertation, Arizona State University, August 1965.

18. Feller, W., An Introduction to Probability Theory and Its Applications, John Wiley and Sons, Inc., New York, 1950.

19. Parzen, E., Stochastic Processes, Holden-Day, Inc., San Francisco, California, 1962.

20. Parzen, E., Stochastic Processes, Holden-Day, Inc., San Francisco, California, 1962, p. 56.

21. 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.

22. Cramer, H., Mathematical Methods of Statistics, Princeton University Press, Princeton, New Jersey, 1945, pp. 248-249.

23. Kendall, M. G., and A. Stuart, The Advanced Theory of Statistics, Vol. 1, New York, Hafner Publishing Company, 1958.

24. Van Slyke, R. M., "Monte Carlo Methods and the PERT Problem," Op. Res., Vol. 11, No. 5, 1963, pp. 839-860.

25. Jury, E. I., Theory and Application of the Z Transform Method, John Wiley and Sons, Inc., New York, 1964.

26. Gupta, S. C., Transform and State Variable Methods in Linear Systems, Text to be published by McGraw-Hill Book Company.

27. Martin, J. J., "Distribution of the Time Through A Directed Acyclic Network," JORSA, V. 13, No. 1, January-February 1965, pp. 46-66.

28. Erdelyi, A., et al., Tables of Integral Transforms, Vol. I, Bateman Manuscript Project, McGraw-Hill Book Company, Inc., New York, 1954.

29. Epstein, B., "Uses of the Mellin Transform in Statistics," Ann. Math. Stat., Vol. 19, 1948, pp. 370-379.