

# Stochastic PERT Networks as Models of Cognition: Derivation of the Mean, Variance, and Distribution of Reaction Time Using Order-of-Processing (OP) Diagrams

DONALD L. FISHER AND WILLIAM M. GOLDSTEIN

*University of Michigan*

June 27, 1983

It is frequently assumed that the mental activity which leads to a given response is made up of separable components or processes. One or more of the processes are assumed to contribute to the time required to respond. Computation of the mean, variance, and distribution of the reaction time is relatively straightforward when all processes are arranged in series or parallel. However, such is not the case when the processes have complex arrangements. A solution to a useful special case of the above problem is proposed. Specifically, it is shown that simple computations yield closed form expressions for the mean, variance, and distribution of reaction time when the processes can be arranged in a stochastic PERT network and when the durations of individual processes are sums of mutually independent, exponentially distributed random variables. The method of solution relies on the construction of an Order-of-Processing (OP) diagram from the original PERT network representation of behavior.

The architecture of the mind is presumably quite complex. Unfortunately, the psychologist cannot peer inside the mind and openly inspect the structure process by process. Instead, the psychologist must make do with a global measure of behavior such as reaction time. It thus becomes of some importance to be able to compute the reaction time distribution associated with a particular arrangement of the mental processes. Summary characteristics of the reaction time distribution such as the mean and variance are also frequently quite useful. This paper will focus on the development of simple, closed form expressions for the mean, variance, and

The authors especially wish to thank J. E. Keith Smith, Stephen Pollock, and Richard Schweickert for their helpful comments and criticisms throughout the writing of this paper. Thanks are also due to Greg Ashby who suggested a way of obtaining a closed form expression for the density function of a stochastic PERT network and to the Editor whose continued interest in the proofs has forced us to think long and hard about alternative approaches. We would also like to thank Clyde H. Coombs for his encouragement and support. The ideas described in this paper were first presented at the meetings of the Society for Mathematical Psychology, Santa Barbara, August 1981 (Note 1). Portions of this research were funded by National Science Foundation grant No. BNS 81-20299 to Clyde H. Coombs, by a Faculty Research grant from the University of Massachusetts, by a National Science Foundation Grant (IST 83-09431), and by a grant from the Horace H. Rackham Graduate School of the University of Michigan. Correspondence and reprint requests should be addressed to Donald L. Fisher, Department of Industrial Engineering and Operations Research, 114 Marston Hall, University of Massachusetts, Amherst, Mass. 01003. William M. Goldstein is now at the University of Chicago.

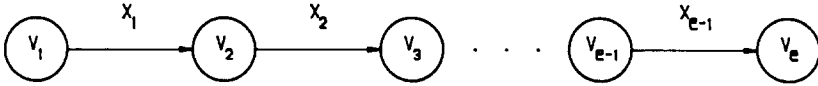


FIG. 1. A representation of mental processes arranged in series. The task is finished when process  $x_{e-1}$  completes executing.

distribution of the duration of behaviors whose constituent processes take on more complex arrangements than have been considered to date.

*The Structure of Cognitive Behavior.* The earliest reaction time models (Donders 1868) of cognition assumed that the behavior of the subjects in various psychological experiments was determined by elementary mental processes which followed one right after another in time (i.e., the mental processes were arranged in series). If the  $i$ th process is labelled  $x_i$ , then a model of processes arranged in series can be represented by the directed, acyclic graph in Fig. 1. Each arc represents a process. Note that  $x$  will serve as a label for the  $i$ th arc and the  $i$ th process; context will be used to clarify any ambiguities which might arise. Vertices  $v_1, v_2, \dots, v_e$  separate the head of one arc from the tail of the next arc. Vertex  $v_1$  will be referred to as the source and vertex  $v_e$  will be referred to as the sink. Process  $x_i$  is said to immediately precede both process  $x_{i+1}$  and vertex  $v_{i+1}$  while process  $x_{i+1}$  is said to immediately succeed both process  $x_i$  and vertex  $v_{i+1}$ .

If the random variable  $X_i$  is defined as the duration of process  $x_i$ , then the duration  $T$  of the task behavior (i.e., the reaction time) has a very simple expression  $T = \sum_{i=1}^n X_i$ . The expected value  $E[T]$  of the task duration  $T$  has an equally simple expression:  $E[T] = \sum_{i=1}^n E[X_i]$ . Unfortunately, closed form characterizations of the distribution of the task duration  $T$  can frequently be difficult to compute. However when the durations of the processes making up a behavior are mutually independent exponentially distributed random variables, then easy-to-compute, closed form expressions for the entire distribution of the task duration can be written down (e.g. Christie and Luce, 1956; McGill, 1963; McGill and Gibbon, 1965).

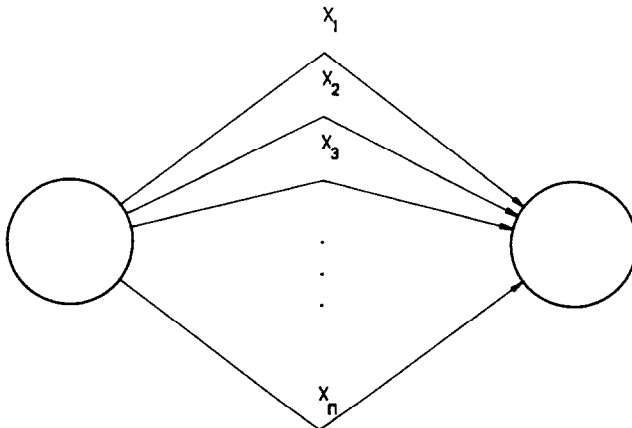


FIG. 2. A representation of mental processes arranged in parallel. The task is completed when the last of the processes  $x_1, \dots, x_n$  completes executing.

A different, but now common arrangement of the mental processes is one in which most (or all) of the processes are executed in parallel (e.g., Atkinson, Holmgren, and Juola, 1969; Egeth, Jonides, and Wall, 1972; Sternberg, 1969; Taylor, 1976; Townsend, 1974). Sometimes it is assumed that the first of the parallel processes connecting two vertices to finish executing terminates activity between these two vertices. At other times it is assumed that the activity continues until all processes connecting the two vertices have been executed (see Fig. 2). That is, it is assumed the task is not finished until each of processes  $x_1, x_2, \dots, x_n$  has completed. The task duration  $T$  is then the maximum of  $X_1, X_2, \dots, X_n$ , while the expected value of  $T$  is the expected value of the maximum of  $X_1, X_2, \dots, X_n$ , i.e.,  $E[T] = E[\max(X_1, X_2, \dots, X_n)]$ . The expected value of the maximum of several random variables is not always easy to compute. However, as above, the computations greatly simplify when the durations of the processes making up a particular behavior are mutually independent, exponentially distributed random variables (Townsend, 1971, 1972).

Recently Schweickert (1978, 1980) has noted that the PERT (Program Evaluation and Review Technique) networks employed widely in operations research can also be used to describe the various arrangements that mental processes might assume. The reader has already been presented with two PERT networks, specifically, with the two graphs in Figs. 1 and 2. A more complex PERT network known as a Wheatstone bridge is represented in Fig. 3. The PERT network is to be interpreted as follows (a more formal definition of the terms used below will be given later in the paper). First, no process can begin executing until *all* processes which immediately precede it have completed their activity. For example, process  $x_5$  cannot begin until processes  $x_3$  and  $x_4$  have completed their activity. Second, once all processes which immediately precede some other process or processes have finished executing, *all* the processes which immediately succeed the immediate predecessors begin their execution. For example, both processes  $x_2$  and  $x_3$  begin executing simultaneously as soon as process  $x_1$  has completed its activity.

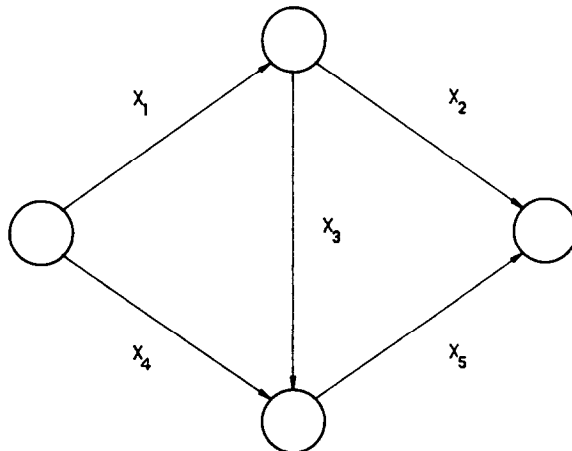


FIG. 3. A Wheatstone bridge PERT network.

An example from the problem solving literature can give psychological substance to a complex PERT network such as a Wheatstone bridge. Suppose subjects are given a syllogism with two premises: "No  $A$  are  $B$ " (the first premise or  $P_1$ ) and "No  $C$  are  $B$ " (the second premise or  $P_2$ ). Suppose that subjects are asked to indicate as quickly as possible whether the conclusion "No  $C$  are  $A$ " is possibly true or necessarily false. The time it takes a subject to respond is the dependent variable.

In order to determine the logical status of the conclusion, the subject must carry out the following activities: (1) read the first premise (READ P1); (2) read the second premise (READ P2); (3) determine the logical meaning of the first premise (ENCODE P1); (4) determine the logical meaning of the second premise (ENCODE P2); (5) decide what set relations between the subject  $C$  and the predicate  $A$  of the conclusion of the syllogism are implied by the first and second premises (COMBINE P1 + P2); (6) read the conclusion (READ CONC); (7) and, finally, decide whether the set relations between  $C$  and  $A$  which follow from the first and second premise are consistent with the statement of the conclusion (READOUT). (A more detailed discussion of the above processes can be found in Fisher, 1981.)

While the above processes could be arranged in a serial network as stated, an extended Wheatstone bridge represents a reasonable alternative arrangement of the processes (see Fig. 4). Subjects start out by reading the first premise of the syllogism (READ P1). After reading the first premise, they not only begin the logical encoding of this premise (ENCODE P1), but also begin reading the second premise (READ P2). Similarly, after reading the second premise subjects not only begin the logical encoding of this premise, but also begin reading the conclusion. Note that subjects cannot combine the information from the first and second premise (COMBINE P1 + P2) until they have read and encoded both premises. Also, note that subjects cannot determine whether a particular conclusion is true (READOUT) until they have both read the conclusion and combined information from the first and second premise.

The "duration" of a path through a PERT network is defined as the sum of the durations of the processes constituting the path. The task duration of a PERT network on a particular trial is then the duration of the longest path through the

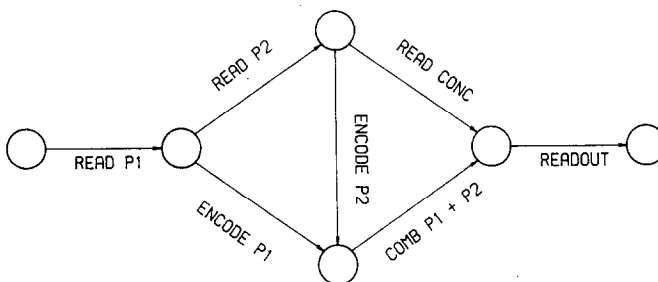


FIG. 4. An extended Wheatstone bridge PERT network used to represent one possible arrangement of the mental processes in a syllogistic reasoning task.

network (e.g., see Schweickert, 1978). As an example, consider the Wheatstone bridge in Fig. 3. There are three paths through this network, the paths represented by arcs  $x_1$  and  $x_2$ , by arcs  $x_1$ ,  $x_3$ , and  $x_5$  and by arcs  $x_4$  and  $x_5$ . The task duration  $T$  on a particular trial is therefore equal to the maximum duration of the three paths through the network, i.e.,

$$T = \text{maximum}(X_1 + X_2, X_1 + X_3 + X_5, X_4 + X_5).$$

The task duration  $T$  is easy enough to compute when the network is deterministic and the process durations  $X_i$  remain constant from trial to trial. But the computations for  $E[T]$  can become very tedious when the process durations vary from trial to trial.

*Stochastic PERT Networks.* When the durations of the processes arranged in a PERT network vary from trial to trial the PERT network is called stochastic. Stochastic PERT networks have been the source of much attention in the operations research literature (e.g., Kleindorfer, 1971; Martin, 1965; Modor and Phillips, 1970; Nadas, 1979; Robillard and Trahan, 1977) ever since an article by Fulkerson (1962) over twenty years ago. Hartley and Worthman (1966) derive exact expressions for the distribution of the task duration when: (1) the constituent process durations are mutually independent, and (2) the PERT network is composed of particular "building block" subnetworks. Ringer (1969) showed how additional subnetworks could serve as "building blocks," and he extended the method so that, in principle, one could find the distribution of task duration for any PERT network whose constituent process durations are mutually independent. However, for both Hartley and Worthman (1966) and Ringer (1969) the resulting expressions are in the form of integrals which may or may not have an analytic solution. While analytic solutions for the expected task duration exist in the case where the density functions of the process durations can be expressed as polynomials with a finite number of coefficients (Martin, 1965), the computations are by no means straightforward.

In this paper a solution to a useful special case of the stochastic PERT problem is proposed. Specifically, it is shown that simple matrix operations yield the mean, variance, and distribution of the task duration when the durations of the constituent processes are sums of mutually independent, exponentially distributed random variables.

*Summary.* One can imagine a sequence of increasingly complex arrangements of the mental processes which mediate behavior in any given task. Relatively simple arrangements include serial and parallel structures. More complex arrangements can be described by PERT networks. (For discussions of still other structures see Note 2; Pritsker and Happ, 1966; Pritsker and Whitehouse, 1966.) As the structures become more complex so too does the computation of the mean, variance, and distribution of the task duration. This paper will focus on the derivation of closed form expressions for the above quantities when the processes are arranged in a PERT network and when the process durations are mutually independent, exponentially distributed random variables.

The remaining portions of this paper are organized as follows. A task system is given a more formal definition. It is shown how the processes in the task system can be arranged in a PERT network. It is then shown how the PERT network itself can receive an alternative representation, a representation which is labelled an Order-of-Processing (OP) diagram. Since the notion of an OP diagram is new with this paper, some time is spent describing such a diagram. Once a PERT network is represented as an OP diagram, computation of the mean, variance, and distribution of the task duration moves along reasonably straightforward lines. In the penultimate section of the paper, the computational methods are extended to cover more complex distributions of process durations. Readers interested in the major results should look only at the next three sections ("Task Systems," "PERT Networks," "Order-of-Processing (OP) Diagrams") and then skip to the subsection ("Example Computations: The Mean") which discusses the use of the OP method to obtain the mean of both a simple parallel network and a Wheatstone bridge network.

### TASK SYSTEMS

Let  $\{x_i\}$  be a finite set of processes. Let  $\mathbf{X}' = (X_1, X_2, \dots, X_n)$  be a random vector of process durations and let  $\mathbf{t}' = (t_1, t_2, \dots, t_n)$  be a vector of constants. Then a task system will be defined as the set  $\{x_i\}$  of  $n$  processes, a strict partial order  $<_R$  ( $<_R$  is an asymmetric, transitive binary relation) on the set of processes, and a known and continuous joint probability density function  $f_{\mathbf{X}'}(\mathbf{t}')$  for the durations  $X_i$  of each of the processes  $x_j$ . Process  $x_i$  will be said to precede process  $x_j$  if  $x_i <_R x_j$  while process  $x_i$  will be said to succeed process  $x_j$  if  $x_j <_R x_i$ . It is assumed that *all* processes in a task system which have no predecessors begin executing simultaneously at the start of the task. It is assumed that a process which has one or more predecessors begins at the moment when *all* of its predecessors have finished executing. Also, it is assumed that once a process has begun execution, it continues uninterrupted until it has finished executing. Finally, it is assumed that a task is not completed until *all* processes in the task have completed their execution. (A more detailed discussion of task systems is given in Coffman, 1976.)

Two other terms should be defined at this point. First, a process  $x_i$  will be defined as an immediate predecessor of a process  $x_j$  if  $x_i <_R x_j$  and if there is no process  $x_k$  such that  $x_i <_R x_k$  and  $x_k <_R x_j$ . For example, suppose that the five processes  $x_1$  through  $x_5$  in a task have the following partial order:  $x_1 <_R x_2$ ,  $x_1 <_R x_3$ ,  $x_3 <_R x_5$ ,  $x_4 <_R x_5$ . Then process  $x_1$  is a predecessor of processes  $x_2$ ,  $x_3$ , and  $x_5$  ( $x_1 <_R x_3$ ,  $x_3 <_R x_5$  imply  $x_1 <_R x_5$ ) but process  $x_1$  is an immediate predecessor of processes  $x_2$  and  $x_3$  only. Second, a process  $x_j$  will be defined as an immediate successor of a process  $x_i$  if  $x_i$  is an immediate predecessor of  $x_j$ . For example, process  $x_5$  is a successor of processes  $x_1$ ,  $x_3$ , and  $x_4$  but is an immediate successor of processes  $x_3$  and  $x_4$  only.

## PERT NETWORKS

It will be more convenient to represent the partial order on the processes in terms of a directed, acyclic graph  $G$  known as a PERT or CPM (Critical Path Method) network. Examples of such networks have already been given (see Figs. 1–4). The networks are constructed from a partial order  $<_R$  on the processes as follows. If process  $x_i$  immediately precedes process  $x_j$  then the head of arc  $x_i$  and the tail of arc  $x_j$  are attached to the same vertex. If there are no predecessors of a process, then the tail of the process is attached to the source, vertex  $v_1$ . If there are no successors of a process, then the head of the arc is attached to the sink, vertex  $v_e$ , where  $e$  equals the number of vertices in the PERT network after all arcs have been drawn and attached. The vertices between the source and the sink are arbitrarily labelled  $v_j$  for  $1 < j < e$ .

Consider as an example the construction of a PERT network from the partial order on the processes given above, i.e.,  $x_1 <_R x_2$ ,  $x_1 <_R x_3$ ,  $x_3 <_R x_5$ ,  $x_4 <_R x_5$ . Since there are no predecessors of processes  $x_1$  and  $x_4$  the tails of the corresponding arcs are attached to vertex  $v_1$ . Next, the head of arc  $x_1$  and the tails of arcs  $x_2$  and  $x_3$  must be attached to the same vertex (since process  $x_1$  immediately precedes processes  $x_2$  and  $x_3$ ), say vertex  $v_2$ , and the head of arcs  $x_3$  and  $x_4$  and the tail of arc  $x_5$  must be attached to the same vertex (since processes  $x_3$  and  $x_4$  immediately precede process  $x_5$ ), say vertex  $v_3$ . Finally, since processes  $x_2$  and  $x_5$  have no successors, the heads of arcs  $x_2$  and  $x_5$  are attached to vertex  $v_4$ . The final result is the Wheatstone bridge network diagrammed in Fig. 3.

Finally, a path through a PERT network should be given a more formal definition. Specifically, a path is defined as a sequence of arcs  $(x_{i_1}, x_{i_2}, \dots, x_{i_k})$  such that the head of arc  $x_{i_j}$  is at the tail of arc  $x_{i_{j+1}}$  for  $j = 1, 2, \dots, k - 1$ . Unless otherwise stated, it will be assumed throughout that the paths under consideration originate at the source and terminate at the sink.

## ORDER-OF-PROCESSING (OP) DIAGRAMS

The order in which the processes begin and complete execution will vary from trial to trial in a stochastic PERT network because the durations of the processes themselves vary from trial to trial. For example, consider the two-process parallel PERT network in Fig. 5a. Processes  $x_1$  and  $x_2$  must begin at the same time. However, on some trials  $x_1$  will finish before  $x_2$  while on the other trials  $x_2$  will finish before  $x_1$ .

*Definition of an OP Diagram.* It will be useful to have a representation for the various possible processing orders associated with a stochastic PERT network. This representation will be labelled an Order-of-Processing diagram, or more simply an OP diagram. The OP diagram associated with the two-process parallel PERT network is drawn in Fig. 5b. Each oval in an OP diagram will be referred to as a state. It will be assumed that there are a total of  $f$  states in an OP diagram. The start and finish states will be labelled, respectively,  $s_1$  and  $s_f$ .

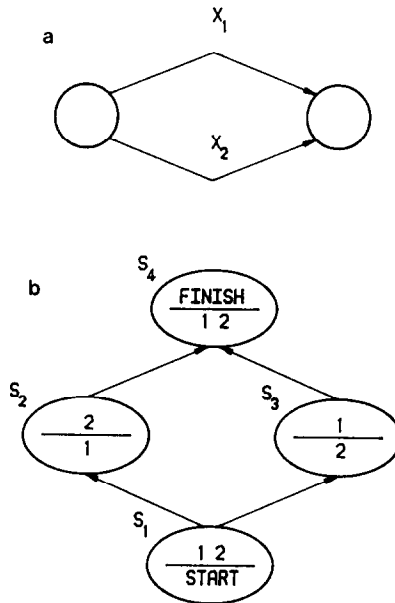


FIG. 5. (a) A two-process parallel PERT network and (b) its associated Order-of-Processing (OP) diagram.

If one were to examine the status of the task system at some particular moment in time, one could distinguish three sets of processes: (1) processes having completed execution; (2) processes currently executing; and (3) processes whose execution has not yet begun. Associated with each state of the OP diagram will be the first two sets of processes, labelled simply the completed and current set. The current set of processes consists of all those processes which are currently executing. These are listed in the upper half of each state. The completed set of processes consists of all those processes which have finished executing. Note that, for simplicity, each process in the current and completed set of a state is represented by the unique integer subscript associated with the process, i.e., process  $x_i$  is labelled as  $i$ .

The states in the OP diagram are connected by directed arcs to one another. A state  $s_j$  is an immediate successor of another state  $s_i$  if a single directed arc from state  $s_i$  to state  $s_j$  joins the two states. A state  $s_i$  is an immediate predecessor of a state  $s_j$  if state  $s_j$  is an immediate successor of state  $s_i$ . A path  $(s_{\alpha_1}, s_{\alpha_2}, \dots, s_{\alpha_{n+1}})$  is a sequence of states from state  $s_1$  to state  $s_f$  such that state  $s_{\alpha_i}$  is an immediate predecessor of state  $s_{\alpha_{i+1}}$ . It should be emphasized that for all paths  $\alpha$ ,  $s_{\alpha_1} = s_1$  and  $s_{\alpha_{n+1}} = s_f$ . Note that in contrast to paths through a PERT network, paths through an OP diagram are always of the same length, i.e., there are  $n + 1$  states in any path. With this notation, any particular path  $(s_{\alpha_1}, s_{\alpha_2}, \dots, s_{\alpha_{n+1}})$  can be identified with a sequence of indices  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_{n+1})$ . It will be convenient to refer to  $\alpha$  as a path, even though  $\alpha$  is a sequence of indices while the corresponding path is a sequence of states. Lower case



Greek letters will be reserved for these index sequences and  $\Gamma = \{\alpha, \beta, \gamma, \dots\}$  will denote the set of all paths.

Each path through an OP diagram represents one possible processing order in a stochastic PERT network. Thus, a new state will be entered along a path from  $s_i$  to  $s_j$  each time a process has completed executing and, simultaneously, some (possibly no) processes have begun executing. For example, consider the OP diagram in Fig. 5. A path is formed by the sequence of states  $(s_1, s_2, s_4)$ . This path defines the following processing order. Processes  $x_1$  and  $x_2$  begin executing simultaneously at the start of a trial because they both appear in the current set of state  $s_1$ . Process  $x_1$  finishes before process  $x_2$  because process  $x_1$  first appears in a completed set (state  $s_2$ ) before process  $x_2$  (state  $s_4$ ). The processing order in which  $x_2$  completes before  $x_1$  is represented by the other path  $(s_1, s_3, s_4)$  through the OP diagram.

Since it was assumed that the distributions of the durations of the processes are continuous and independent, the probability that processes  $x_1$  and  $x_2$  complete at exactly the same time is zero. This means that a transition directly from state  $s_1$  to state  $s_4$  in Fig. 5 has probability zero. It will turn out that paths of probability zero such as the one from state  $s_1$  to state  $s_4$  can be ignored when computing the expected task duration. Thus, rather than clutter up the OP diagram with an arc which will not figure in the computations, no arc is drawn between states  $s_1$  and  $s_4$ . More generally, since the probability that two or more processes finish at exactly the same time is zero, no arcs are drawn between any two states  $s_i$  and  $s_j$  in an OP diagram which require more than the completion of one process in the current set of state  $s_i$  to move the system from state  $s_i$  to state  $s_j$ .

It may be helpful to consider a second, more complex OP diagram. In particular, consider an OP diagram (Fig. 6) which is constructed from a Wheatstone bridge stochastic PERT network (Fig. 3). The path through the left-hand side of the OP diagram consists of the sequence of states  $(s_1, s_2, s_4, s_7, s_{10}, s_{12})$ . Along this path, processes  $x_1$  and  $x_4$  begin executing simultaneously at the start of a task since they appear in the current set of the start state. Process  $x_4$  is the first process to complete along the path since it appears in the completed set of the next state along the path, state  $s_2$ . No new processes begin when  $x_4$  completes since all processes in the current set of state  $s_2$  have previously begun executing. Process  $x_1$  is the second process to complete along the path since it appears for the first time in the completed set of the next state, state  $s_4$ . Note that the completion of process  $x_1$  is followed immediately by the start of the execution of processes  $x_2$  and  $x_3$  since they appear in a current set for the first time in state  $s_4$ . The interpretation of Fig. 6 continues in this fashion.

*Additional Notation.* The random variable  $T_j$  will be used to indicate the duration of state  $s_j$ . The random variable  $T$  will continue to indicate the task duration. The random variable  $X_k$  will continue to indicate the duration of process  $x_k$ .

Consider a path  $\alpha$  consisting of states  $(s_{\alpha_1}, s_{\alpha_2}, \dots, s_{\alpha_{n+1}})$ . The system moves from state  $s_{\alpha_j}$  to state  $s_{\alpha_{j+1}}$  when a certain process, say  $x_{\alpha_j^*}$ , is completed. Let  $\alpha^*$  be the sequence  $(\alpha_1^*, \alpha_2^*, \dots, \alpha_n^*)$ . Thus, process  $x_{\alpha_j^*}$  is the  $j$ th process to be completed if the system takes path  $\alpha$  through the OP diagram. Since every process must be completed

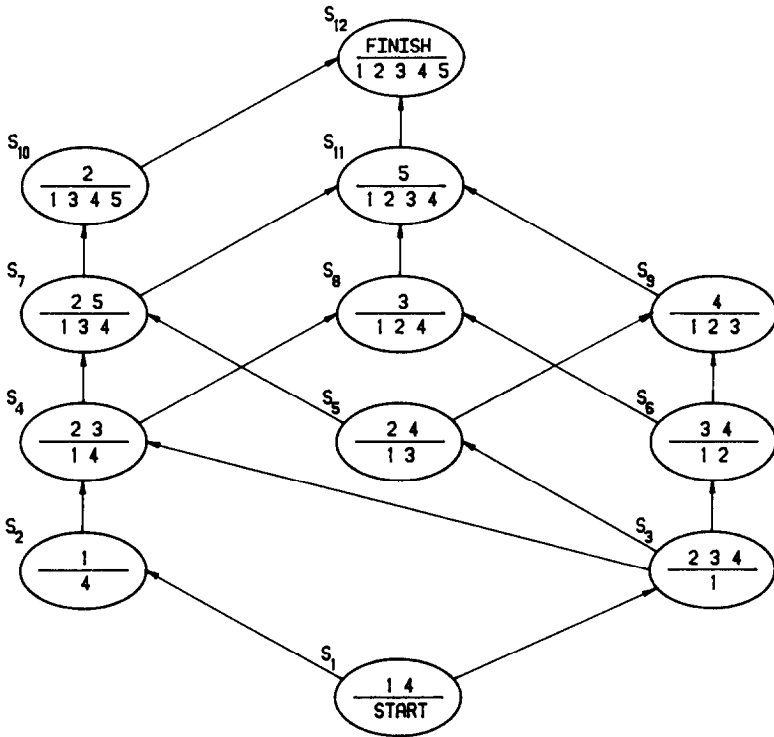


FIG. 6. The Order-of-Processing (OP) diagram for the stochastic Wheatstone bridge PERT network represented in Fig. 3.

for the final state to be reached, every process receives an index  $\alpha_j^*$  for some  $j$ ,  $1 \leq j \leq n$ .

Finally, let  $P_\alpha$  be an indicator random variable which equals 1 if path  $\alpha$  is taken and which equals 0 otherwise.

PERT PROCESS DURATIONS AND OP STATE DURATIONS

Two separate representations for a task system have been presented: the OP diagram and the PERT network. Relations between quantities in the OP diagram and quantities in the PERT network will now be derived. Specifically, it will be shown that the state duration  $T_{\alpha_i}$  can be expressed as a linear combination  $Z_{\alpha_i}$  of the process durations  $X_j$ , given that a particular path  $\alpha$  in the OP diagram is followed (Lemma 1). Also, it will be shown that if  $Z_{\alpha_j} > 0$  for  $j = 1, 2, \dots, n$ , then path  $\alpha$  must have been taken (Lemma 2). Finally, it will be shown that the relatively complex linear combination  $Z_{\alpha_i}$  of the process durations  $X_j$  can be computed quite simply when reformulated in matrix terms.

*Notation.* Define a square  $n$  by  $n$  matrix  $A_\alpha$  as consisting of components  $a_{ij}$ , where

$$\begin{aligned} a_{ij} &= 1 && \text{if process } x_{\alpha_i^*} \text{ is in the current set of state } s_{\alpha_j}, \\ a_{ij} &= 0 && \text{otherwise.} \end{aligned}$$

Define the random variables  $Z_{\alpha_i}$  recursively as

$$\begin{aligned} Z_{\alpha_1} &= X_{\alpha_1^*}, \\ Z_{\alpha_i} &= X_{\alpha_i^*} - \sum_{j=1}^{i-1} a_{ij} Z_{\alpha_j} \quad \text{for } i = 2, 3, \dots, n. \end{aligned}$$

Let both  $Z'_\alpha = (Z_{\alpha_1}, \dots, Z_{\alpha_n})$  and  $X'_\alpha = (X_{\alpha_1^*}, \dots, X_{\alpha_n^*})$  be random vectors. Finally, it will be convenient to have the notation  $T'_\alpha = (T_{\alpha_1}, \dots, T_{\alpha_n})$ .

LEMMA 1. *If path  $\alpha$  is followed, then*

$$T_{\alpha_i} = Z_{\alpha_i} \quad \text{and} \quad T_{\alpha_i}, Z_{\alpha_i} > 0.$$

*Proof.* If  $P_\alpha = 1$ , then the duration  $T_{\alpha_i}$  of state  $s_{\alpha_i}$  is positive and equal to the duration  $X_{\alpha_i^*}$  of the process which gets completed in state  $s_{\alpha_i}$  minus its duration in all the preceding states in which it was current, i.e.,

$$T_{\alpha_i} = X_{\alpha_i^*} - \sum_{j=1}^{i-1} a_{ij} T_{\alpha_j},$$

from which the proof follows easily by induction.

Note that the reason for introducing the random variable  $Z_{\alpha_i}$  is that it will be useful to have an expression for the linear combination of the  $X_{\alpha_i^*}$ 's in cases where the path  $\alpha$  is not taken, i.e., in cases where the linear combinations may possibly take on negative values and therefore where  $T_{\alpha_i}$  does not equal the linear combination.

LEMMA 2. *If  $Z_{\alpha_i} > 0$  for  $i = 1, 2, \dots, n$ , then  $P_\alpha = 1$ .*

*Proof.* Suppose that  $Z_{\alpha_i} > 0$  for  $i = 1, 2, \dots, n$ . Let  $C_{\alpha_i} = \{\alpha_k^* \mid x_{\alpha_k^*} \text{ is current in } s_{\alpha_i}\}$  be the set of indices of the processes which are current in state  $s_{\alpha_i}$ .

First, it will be shown that states  $s_{\alpha_1}$  and  $s_{\alpha_2}$  are both reached and that  $T_{\alpha_1} = Z_{\alpha_1} = X_{\alpha_1^*}$ . The proof that state  $s_{\alpha_1}$  is reached is immediate since there is only one start state, namely  $s_{\alpha_1}$ . The transition is made to state  $s_{\alpha_2}$  if and only if  $X_{\alpha_1^*} < X_{\alpha_k^*}$  for all  $\alpha_k^*$  in  $C_{\alpha_1} - \{\alpha_1^*\}$ . Let  $\alpha_k^*$  be in  $C_{\alpha_1} - \{\alpha_1^*\}$ . Then  $k > 1$ . Now,

$$\begin{aligned} 0 < Z_{\alpha_k} &= X_{\alpha_k^*} - \sum_{j=1}^{k-1} a_{kj} Z_{\alpha_j} \\ &\leq X_{\alpha_k^*} - a_{k1} Z_{\alpha_1} \\ &= X_{\alpha_k^*} - a_{k1} X_{\alpha_1^*}. \end{aligned}$$

Since  $\alpha_k^*$  is in  $C_{\alpha_i} - \{\alpha_i^*\}$ , it is the case that  $a_{ki} = 1$  and therefore  $X_{\alpha_i} < X_{\alpha_k}$  as needed. Furthermore it is clear that  $T_{\alpha_i} = Z_{\alpha_i} = X_{\alpha_i}$ .

Second, given the hypothesis of Lemma 2, it will be shown that (1) if states  $s_{\alpha_1}, \dots, s_{\alpha_i}$  are all reached and (2) if  $T_{\alpha_j} = Z_{\alpha_j}$  for  $j = 1, \dots, i-1$ , then  $s_{\alpha_{i+1}}$  is reached and  $T_{\alpha_i} = Z_{\alpha_i}$ . Note that the transition is made from  $s_{\alpha_i}$  to  $s_{\alpha_{i+1}}$  if and only if

$$0 < X_{\alpha_i} - \sum_{j=1}^{i-1} a_{ij} T_{\alpha_j} < X_{\alpha_k} - \sum_{j=1}^{i-1} a_{kj} T_{\alpha_j}$$

for all  $\alpha_k^*$  in  $C_{\alpha_i} - \{\alpha_i^*\}$ . Moreover, if this occurs, it is clear that

$$Z_{\alpha_i} = X_{\alpha_i} - \sum_{j=1}^{i-1} a_{ij} Z_{\alpha_j} = X_{\alpha_i} - \sum_{j=1}^{i-1} a_{ij} T_{\alpha_j} = T_{\alpha_i},$$

since, by hypothesis,  $T_{\alpha_j} = Z_{\alpha_j}$  for  $j = 1, 2, \dots, i-1$ .

Let  $\alpha_k^*$  be in  $C_{\alpha_i} - \{\alpha_i^*\}$ . Then  $k > i$ . By hypothesis,  $Z_{\alpha_k} > 0$ . Thus,

$$\begin{aligned} 0 < Z_{\alpha_k} &= X_{\alpha_k} - \sum_{j=1}^{k-1} a_{kj} Z_{\alpha_j} \\ &= X_{\alpha_k} - \sum_{j=1}^{i-1} a_{kj} Z_{\alpha_j} - \sum_{j=i}^{k-1} a_{kj} Z_{\alpha_j} \\ &= X_{\alpha_k} - \sum_{j=1}^{i-1} a_{kj} T_{\alpha_j} - \sum_{j=i}^{k-1} a_{kj} Z_{\alpha_j} \\ &\leq X_{\alpha_k} - \sum_{j=1}^{i-1} a_{kj} T_{\alpha_j} - a_{ki} Z_{\alpha_i}. \end{aligned}$$

Since  $\alpha_k^*$  is in  $C_{\alpha_i} - \{\alpha_i^*\}$ , it must be the case that  $a_{ki} = 1$ . Thus, after rearranging the last inequality, one obtains

$$0 < Z_{\alpha_i} = X_{\alpha_i} - \sum_{j=1}^{i-1} a_{ij} T_{\alpha_j} < X_{\alpha_k} - \sum_{j=1}^{i-1} a_{kj} T_{\alpha_j},$$

as required. This completes the proof of Lemma 2.

**LEMMA 3.** For any path  $\alpha$ ,  $\det \mathbf{A}_\alpha = \det \mathbf{A}_\alpha^{-1} = 1$ .

*Proof.* By construction,  $\mathbf{A}_\alpha$  is a lower triangular matrix with 1's on the diagonal. This follows since by definition process  $x_{\alpha_i}$  is completed before state  $s_{\alpha_{i+k}}$  is entered and therefore  $a_{i,i+k} = 0$  for  $1 \leq k \leq n-i$  and since by definition process  $x_{\alpha_i}$  is completed in state  $s_{\alpha_i}$  and therefore  $a_{ii} = 1$ . Thus  $\det \mathbf{A}_\alpha = 1$ .

Since  $\det \mathbf{A}_\alpha \neq 0$ ,  $\mathbf{A}_\alpha^{-1}$  exists. Furthermore, since  $1 = \det \mathbf{I} = \det[\mathbf{A}_\alpha][\mathbf{A}_\alpha^{-1}] = [\det \mathbf{A}_\alpha][\det \mathbf{A}_\alpha^{-1}]$ , it follows that  $\det \mathbf{A}_\alpha^{-1} = 1/\det \mathbf{A}_\alpha = 1$ . This completes the proof.

LEMMA 4.

$$\mathbf{Z}_\alpha = \mathbf{A}_\alpha^{-1} \mathbf{X}_{\alpha^*}.$$

*Proof.* It will be shown that  $\mathbf{A}_\alpha \mathbf{Z}_\alpha = \mathbf{X}_{\alpha^*}$ , from which Lemma 4 follows immediately. The  $i$ th component of  $\mathbf{A}_\alpha \mathbf{Z}_\alpha$  is,

$$\begin{aligned} \sum_{j=1}^n a_{ij} Z_{\alpha_j} &= \sum_{j=1}^i a_{ij} Z_{\alpha_j} \quad \text{since } a_{ij} = 0 \text{ for } j > i, \\ &= a_{ii} Z_{\alpha_i} + \sum_{j=1}^{i-1} a_{ij} Z_{\alpha_j} \\ &= Z_{\alpha_i} + \sum_{j=1}^{i-1} a_{ij} Z_{\alpha_j} \quad \text{since } a_{ii} = 1, \\ &= X_{\alpha_i^*} - \sum_{j=1}^{i-1} a_{ij} Z_{\alpha_j} + \sum_{j=1}^{i-1} a_{ij} Z_{\alpha_j}, \end{aligned}$$

by definition of  $Z_{\alpha_i}$ ,

$$= X_{\alpha_i^*}.$$

This completes the proof of Lemma 4.

LEMMA 5.  $P_\alpha = 1$  if and only if  $\mathbf{A}_\alpha^{-1} \mathbf{X}_{\alpha^*} > \mathbf{0}$ , i.e.,  $P_\alpha = 1$  if and only if each component of  $\mathbf{A}_\alpha^{-1} \mathbf{X}_{\alpha^*}$  is positive.

*Proof.* If  $P_\alpha = 1$ , then by Lemma 1  $\mathbf{T}_\alpha = \mathbf{Z}_\alpha > \mathbf{0}$ . By Lemma 4,  $\mathbf{Z}_\alpha = \mathbf{A}_\alpha^{-1} \mathbf{X}_{\alpha^*}$  and therefore if  $P_\alpha = 1$ ,  $\mathbf{A}_\alpha^{-1} \mathbf{X}_{\alpha^*} > \mathbf{0}$ .

If  $\mathbf{A}_\alpha^{-1} \mathbf{X}_{\alpha^*} > \mathbf{0}$ , then since by Lemma 4  $\mathbf{Z}_\alpha = \mathbf{A}_\alpha^{-1} \mathbf{X}_{\alpha^*}$ , it follows by Lemma 2 that  $P_\alpha = 1$ .

#### DISTRIBUTION OF TASK DURATION: EXPONENTIAL CASE

In this section it is assumed that the process durations are mutually independent, exponentially distributed random variables. That is,

$$\begin{aligned} f_{\mathbf{X}_{\alpha^*}}(\mathbf{t}') &= \prod_{i=1}^n \lambda_{\alpha_i^*} e^{-\lambda_{\alpha_i^*} t_i} \quad \text{if } t_1, \dots, t_n > 0, \\ &= 0 \quad \text{otherwise.} \end{aligned}$$

The quantity  $\lambda_{\alpha_i^*}$  will be referred to as the rate parameter for  $X_{\alpha_i^*}$ . Closed form expressions for the mean, variance, and distribution of the task duration will be derived.

### Mean and Variance

The method for deriving the mean and variance relies on the following observation  $T = \sum_{i=1}^{f-1} T_i$ . This equation follows since on any particular trial  $T_i > 0$  if state  $s_i$  is entered for  $i = 1, 2, \dots, f-1$ , while  $T_i = 0$  otherwise. Note that the duration  $T_f$  of the final state is equal to zero since the task is completed as soon as the final state is entered. For example, consider the OP diagram in Fig. 6. If the left-hand path is taken then the state durations  $T_1, T_2, T_4, T_7$ , and  $T_{10}$  are positive while all other state durations are zero.

If the indicator random variable  $S_i$  is defined as below,

$$\begin{aligned} S_i &= 1 && \text{if state } s_i \text{ is entered, } && i = 1, \dots, f, \\ &= 0 && \text{otherwise, } && i = 1, \dots, f, \end{aligned}$$

then the expected task duration  $E[T]$  can be written as a function of the conditional expectation  $E[T_i | S_i = 1]$  of  $T_i$  given that state  $s_i$  is entered and the probability  $P(S_i = 1)$  that state  $s_i$  is entered. Specifically,

$$\begin{aligned} E[T] &= \sum_{i=1}^{f-1} E[T_i] \\ &= \sum_{i=1}^{f-1} \{E[T_i | S_i = 1] P(S_i = 1) + E[T_i | S_i = 0] P(S_i = 0)\} \\ &= \sum_{i=1}^{f-1} E[T_i | S_i = 1] P(S_i = 1), \end{aligned} \quad (1)$$

since  $T_i = 0$  if  $S_i = 0$ .

A formula for the variance  $\text{VAR}(T)$  of the task duration follows in a similar fashion:

$$\begin{aligned} \text{VAR}(T) &= \sum_{i=1}^{f-1} \text{VAR}[T_i] + 2 \sum_{i < j} \text{COV}[T_i, T_j] \\ &= \sum_{i=1}^{f-1} \{E[T_i^2] - E[T_i]^2\} \\ &\quad + 2 \sum_{i < j} \{E[T_i T_j] - E[T_i] E[T_j]\} \\ &= \sum_{i=1}^{f-1} \{E[T_i^2 | S_i = 1] P(S_i = 1) - (E[T_i | S_i = 1] P(S_i = 1))^2\} \\ &\quad + 2 \sum_{i < j} \{E[T_i T_j | S_i = 1, S_j = 1] P(S_i = 1, S_j = 1) \\ &\quad - E[T_i | S_i = 1] P(S_i = 1) E[T_j | S_j = 1] P(S_j = 1)\}. \end{aligned} \quad (2)$$

The above expressions for the mean and variance make clear the need to find formulas for the following quantities:  $P(S_i = 1)$ ,  $P(S_i = 1, S_j = 1)$ ,  $f_{T_i|S_i=1}(t)$ , and  $f_{T_i, T_j|S_i=1, S_j=1}(t_1, t_2)$ . The computation of these quantities is most easily obtained by deriving expressions both for the probability  $P(P_\alpha = 1)$  that path  $\alpha$  is taken (Lemma 7) and for the conditional density  $f_{T_\alpha|P_\alpha=1}(\mathbf{t}')$  of the times spent in each of the states along a path  $\alpha$ , given that the path  $\alpha$  was taken (Lemma 8).

LEMMA 6. Let  $\mathbf{z}' = (z_1, \dots, z_n)$ . Then the density function for  $\mathbf{Z}_\alpha = \mathbf{A}_\alpha^{-1}\mathbf{X}_\alpha$  is given by

$$\begin{aligned} f_{\mathbf{Z}_\alpha}(\mathbf{z}') &= f_{\mathbf{X}_\alpha} \cdot [(\mathbf{A}_\alpha \mathbf{z})'] && \text{if } \mathbf{z} \text{ in set } B_\alpha, \\ &= 0 && \text{if } \mathbf{z} \text{ not in set } B_\alpha, \end{aligned}$$

where  $B_\alpha = \{(z_1, \dots, z_n) \mid \mathbf{z} = \mathbf{A}_\alpha^{-1}\mathbf{y} \text{ and } f_{\mathbf{X}_\alpha}(\mathbf{y}') > 0 \text{ for some } \mathbf{y}\}$ .

*Proof.* A well-known result in probability theory states that

$$\begin{aligned} f_{\mathbf{Z}_\alpha}(\mathbf{z}') &= (1/|\det \mathbf{A}_\alpha^{-1}|) f_{\mathbf{X}_\alpha} \cdot [(\mathbf{A}_\alpha \mathbf{z})'] && \text{if } \mathbf{z} \text{ in set } B_\alpha, \\ &= 0 && \text{if } \mathbf{z} \text{ is not in set } B_\alpha. \end{aligned}$$

The result from Lemma 3 that  $\det \mathbf{A}_\alpha^{-1} = 1$  completes the proof.

LEMMA 7.

$$P(P_\alpha = 1) = \int_{z_n > 0} \cdots \int_{z_1 > 0} f_{\mathbf{X}_\alpha} \cdot [(\mathbf{A}_\alpha \mathbf{z})'] dz_1, \dots, dz_n.$$

*Proof.* Using Lemma 5, one can write

$$P(P_\alpha = 1) = \int_{z_n > 0} \cdots \int_{z_1 > 0} f_{\mathbf{Z}_\alpha}(\mathbf{z}') dz_1, \dots, dz_n.$$

Suppose  $z_1, \dots, z_n > 0$  and consider  $\mathbf{y} = \mathbf{A}_\alpha \mathbf{z}$ . It has been observed that  $\mathbf{A}_\alpha$  is lower triangular with 1's on the diagonal, and by its construction  $\mathbf{A}_\alpha$  contains no negative elements. Thus each component of  $\mathbf{A}_\alpha \mathbf{z}$  is positive. Having assumed mutually independent, exponentially distributed process durations, it follows that the density function  $f_{\mathbf{X}_\alpha}(\mathbf{A}_\alpha \mathbf{z})$  is greater than zero. Furthermore, since  $\mathbf{A}_\alpha$  has an inverse,  $\mathbf{z}$  can always be written as  $\mathbf{A}_\alpha^{-1}\mathbf{y}$ . Thus, if  $z_1, \dots, z_n > 0$  then  $\mathbf{z} \in B_\alpha$ . The desired result then follows immediately from Lemma 6.

LEMMA 8. The conditional density of  $\mathbf{T}_\alpha$  given that  $P_\alpha = 1$  is

$$\begin{aligned} f_{\mathbf{T}_\alpha|P_\alpha=1}(\mathbf{t}') &= [P(P_\alpha = 1)]^{-1} f_{\mathbf{Z}_\alpha}(\mathbf{t}') && \text{if } \mathbf{t} > \mathbf{0}, \\ &= 0 && \text{otherwise.} \end{aligned}$$

*Proof.* It will be shown that the event  $E_1 = [(T_{\alpha_1} \leq t_1), \dots, (T_{\alpha_n} \leq t_n), (P_\alpha = 1)]$  is equivalent to the event  $E_2 = [(0 < Z_{\alpha_1} \leq t_1), \dots, (0 < Z_{\alpha_n} \leq t_n)]$ .

Since, by definition,  $P_\alpha = 1$  is equivalent to  $0 < T_{\alpha_i}$  for  $i = 1, 2, \dots, n$ , event  $E_1$  is equivalent to  $0 < T_{\alpha_i} \leq t_i$  for  $i = 1, 2, \dots, n$ . By Lemma 1, this implies event  $E_2$ .

Now suppose that event  $E_2$  occurs. Since each component of  $Z_\alpha$  is positive, by Lemma 2 it is the case that  $P_\alpha = 1$ . This implies by Lemma 1 that  $Z_\alpha = T_\alpha$ . Thus, event  $E_2$  implies  $[(0 < T_{\alpha_1} \leq t_1), \dots, (0 < T_{\alpha_n} \leq t_n), (P_\alpha = 1)]$ . However,  $P_\alpha = 1$  is redundant with  $0 < T_{\alpha_i}$  for all  $i$ . Thus, event  $E_2$  implies  $[(T_{\alpha_1} \leq t_1), \dots, (T_{\alpha_n} \leq t_n), (P_\alpha = 1)]$ , i.e., event  $E_2$  implies event  $E_1$ . This completes the proof of Lemma 8.

Before proceeding further with the derivation of closed form expressions for the mean and variance, three additional bits of notation will be useful. Let

$$\begin{aligned} \lambda_{ij} = \lambda_k & \quad \text{if the completion of process } x_k \\ & \quad \text{in state } s_i \text{ moves the system immediately} \\ & \quad \text{to state } s_j, \\ = 0 & \quad \text{otherwise (i.e., if there is no process whose} \\ & \quad \text{completion in state } s_i \text{ moves the system immediately} \\ & \quad \text{to state } s_j). \end{aligned}$$

Let  $c_i = \sum_{j=1}^f \lambda_{ij}$  be the sum of the rate parameters of the processes current in state  $s_i$  and let  $c_{\alpha_i} = \sum_{j=1}^n a_{ji} \lambda_{\alpha_j}$  be the sum of the rate parameters of the processes current in the  $i$ th state reached along path  $\alpha$ . (Note that  $c_i \neq c_{\alpha_i}$  unless  $s_i = s_{\alpha_i}$ .) Finally, let  $\mathbf{D}$  be an  $f$  by  $f$  matrix whose  $(i, j)$  component is given by  $d_{ij} = (\lambda_{ij}/c_i)$ .

#### THEOREM 1.

$$\begin{aligned} f_{T_\alpha | P_\alpha = 1}(\mathbf{t}') &= [P(P_\alpha = 1)]^{-1} \prod_{j=1}^n \lambda_{\alpha_j} \exp[-c_{\alpha_j} t_j], \quad \mathbf{t} > \mathbf{0}, \\ &= 0 \quad \text{otherwise.} \end{aligned}$$

*Proof.* By Lemma 8,

$$\begin{aligned} f_{T_\alpha | P_\alpha = 1}(\mathbf{t}') &= [P(P_\alpha = 1)]^{-1} f_{Z_\alpha}(\mathbf{t}') \quad \text{provided } \mathbf{t} > \mathbf{0}, \\ &= [P(P_\alpha = 1)]^{-1} f_{X_\alpha^*}[(\mathbf{A}_\alpha \mathbf{t}')] \quad \text{by Lemma 6,} \\ &= [P(P_\alpha = 1)]^{-1} f_{X_\alpha^*} \left[ \sum_{j=1}^n a_{1j} t_j, \dots, \sum_{j=1}^n a_{nj} t_j \right] \\ &= [P(P_\alpha = 1)]^{-1} \prod_{i=1}^n \lambda_{\alpha_i} \exp \left[ -\lambda_{\alpha_i} \sum_{j=1}^n a_{ij} t_j \right] \\ &= [P(P_\alpha = 1)]^{-1} \left( \prod_{i=1}^n \lambda_{\alpha_i} \right) \left( \prod_{i=1}^n \exp \left[ -\lambda_{\alpha_i} \sum_{j=1}^n a_{ij} t_j \right] \right) \end{aligned}$$



$$\begin{aligned}
 &= [P(P_\alpha = 1)]^{-1} \left( \prod_{i=1}^n \lambda_{\alpha_i^*} \right) \left( \exp \left[ \sum_{i=1}^n \sum_{j=1}^n -\lambda_{\alpha_i^*} a_{ij} t_j \right] \right) \\
 &= [P(P_\alpha = 1)]^{-1} \left( \prod_{i=1}^n \lambda_{\alpha_i^*} \right) \left( \exp \left[ \sum_{j=1}^n \sum_{i=1}^n -\lambda_{\alpha_i^*} a_{ij} t_j \right] \right) \\
 &= [P(P_\alpha = 1)]^{-1} \left( \prod_{i=1}^n \lambda_{\alpha_i^*} \right) \left( \exp \left[ \sum_{j=1}^n t_j \sum_{i=1}^n -\lambda_{\alpha_i^*} a_{ij} \right] \right) \\
 &= [P(P_\alpha = 1)]^{-1} \left( \prod_{i=1}^n \lambda_{\alpha_i^*} \right) \left( \prod_{j=1}^n \exp \left[ t_j \sum_{i=1}^n -\lambda_{\alpha_i^*} a_{ij} \right] \right) \\
 &= [P(P_\alpha = 1)]^{-1} \prod_{j=1}^n \lambda_{\alpha_j^*} \exp \left[ - \left( \sum_{i=1}^n \lambda_{\alpha_i^*} a_{ij} \right) t_j \right] \\
 &= [P(P_\alpha = 1)]^{-1} \prod_{j=1}^n \lambda_{\alpha_j^*} \exp[-c_{\alpha_j} t_j].
 \end{aligned}$$

This completes the proof of Theorem 1.

**THEOREM 2.**

$$P(P_\alpha = 1) = \prod_{j=1}^n (\lambda_{\alpha_j^*} / c_{\alpha_j}).$$

*Proof.* By Lemma 7,

$$P(P_\alpha = 1) = \int_{z_n > 0} \cdots \int_{z_1 > 0} f_{X_\alpha^*}[(\mathbf{A}_\alpha \mathbf{z})'] dz_1, \dots, dz_n.$$

By the proof of Theorem 1,

$$f_{X_\alpha^*}[(\mathbf{A}_\alpha \mathbf{z})'] = \prod_{j=1}^n \lambda_{\alpha_j^*} \exp[-c_{\alpha_j} z_j].$$

Substituting and performing the integration, one obtains

$$P(P_\alpha = 1) = \prod_{j=1}^n (\lambda_{\alpha_j^*} / c_{\alpha_j}).$$

This completes the proof of Theorem 2.

**COROLLARY 1.**

$$f_{T_\alpha | P_\alpha = 1}(\mathbf{t}') = \prod_{j=1}^n c_{\alpha_j} \exp[-c_{\alpha_j} t_j], \quad \mathbf{t} > \mathbf{0},$$

*Proof.* By Theorem 1,

$$\begin{aligned} f_{T_\alpha | P_\alpha = 1}(\mathbf{t}') &= [P(P_\alpha = 1)]^{-1} \prod_{j=1}^n \lambda_{\alpha_j} \exp[-c_{\alpha_j} t_j] \\ &= \left[ \prod_{j=1}^n (\lambda_{\alpha_j} / c_{\alpha_j}) \right]^{-1} \left( \prod_{j=1}^n \lambda_{\alpha_j} \right) \left[ \prod_{j=1}^n \exp(-c_{\alpha_j} t_j) \right], \end{aligned}$$

by Theorem 2, and after regrouping

$$= \prod_{j=1}^n c_{\alpha_j} \exp(-c_{\alpha_j} t_j).$$

This completes the proof of Corollary 1.

COROLLARY 2.

$$f_{T_{\alpha_j} | P_\alpha = 1}(t) = c_{\alpha_j} \exp[-c_{\alpha_j} t].$$

*Proof.* Integrating  $f_{T_\alpha | P_\alpha = 1}(\mathbf{t}')$  in Corollary 1 over all  $t_k$  except  $t_k = t_j$  gives the above density. This completes the proof of Corollary 2.

It follows immediately that the conditional distributions of the state durations  $T_{\alpha_j}$  are mutually independent and exponential.

LEMMA 9.

$$f_{T_i | S_i = 1}(t) = c_i \exp[-c_i t].$$

*Proof.* Suppose state  $s_i$  is the  $j$ th state reached along some particular path. Then state  $s_i$  will be the  $j$ th state reached on any path which contains  $s_i$ . Consider the sum over all paths  $\alpha$  in which the  $j$ th component of  $\alpha$  is equal to  $i$ . This sum will be denoted  $\sum_{\alpha_j = i}$ . The probability  $P(S_i = 1)$  that state  $s_i$  is entered can then be expressed as the sum  $\sum_{\alpha_j = i} P(P_\alpha = 1)$ , i.e., the sum of the probabilities of those paths  $\alpha$  in the OP diagram which pass through state  $s_i$ . The conditional density of  $T_i$ , given state  $s_i$  is entered, can now be rewritten as

$$\begin{aligned} f_{T_i | S_i = 1}(t) &= [P(S_i = 1)]^{-1} \sum_{\alpha_j = i} f_{T_{\alpha_j} | P_\alpha = 1}(t) P(P_\alpha = 1) \\ &= [P(S_i = 1)]^{-1} \sum_{\alpha_j = i} [c_{\alpha_j} \exp(-c_{\alpha_j} t)] P(P_\alpha = 1), \\ &= [P(S_i = 1)]^{-1} c_i \exp(-c_i t) \sum_{\alpha_j = i} P(P_\alpha = 1) \\ &= c_i \exp(-c_i t) \left[ \sum_{\alpha_j = i} P(P_\alpha = 1) \right] [P(S_i = 1)]^{-1} \\ &= c_i \exp(-c_i t). \end{aligned}$$

This completes the proof of Lemma 9.

COROLLARY 3.  $E[T_i | S_i = 1] = 1/c_i$  and  $E[T_i^2 | S_i = 1] = 2/c_i^2$ .

*Proof.* The corollary follows immediately from Lemma 9.

LEMMA 10.  $f_{T_i, T_j | S_i=1, S_j=1}(t_1, t_2) = [c_i \exp(-c_i t_1)][c_j \exp(-c_j t_2)]$ .

*Proof.* The derivation is similar to Lemma 9.

COROLLARY 4.  $E[T_i T_j | S_i = 1, S_j = 1] = 1/(c_i c_j)$ .

*Proof.* The corollary follows immediately from Lemma 10.

Recall that  $\mathbf{D}$  is an  $f \times f$  matrix whose  $(i, j)$  component is given by  $d_{ij} = (\lambda_{ij}/c_i)$ .

LEMMA 11. *Suppose state  $s_i$  has  $q - 1$  processes in its completed set. Then*

$$P(S_i = 1) = \mathbf{e}'_i \mathbf{D}^{q-1} \mathbf{e}_i,$$

where  $\mathbf{e}_i$  is an  $f$ -dimensional column vector with a 1 as its  $i$ th component and 0's everywhere else.

*Proof.* Consider a sequence  $(k_1, k_2, \dots, k_{n+1})$ , where  $k_j \in \{1, 2, \dots, f\}$  for  $(1 \leq j \leq n + 1)$ . The sequence  $(k_1, k_2, \dots, k_{n+1})$  may or may not be identical to some sequence of state indices  $\alpha$ . In any case, by definition we have

$$\prod_{j=1}^n \lambda_{k_j k_{j+1}} / c_{k_j} = \prod_{j=1}^n d_{k_j k_{j+1}}. \quad (3)$$

If the sequence  $(k_1, k_2, \dots, k_{n+1})$  of indices is identical to some sequence  $\alpha$ , then by Theorem 2 the above product gives the probability of taking path  $\alpha$ , i.e.,

$$\begin{aligned} P(P_\alpha = 1) &= \prod_{j=1}^n (\lambda_{\alpha_j} / c_{\alpha_j}) \\ &= \prod_{j=1}^n (\lambda_{k_j k_{j+1}} / c_{k_j}) \\ &= \prod_{j=1}^n d_{k_j k_{j+1}}. \end{aligned}$$

However, if the sequence  $(k_1, k_2, \dots, k_{n+1})$  of indices is not identical to any sequence  $\alpha$ , then  $\lambda_{k_j k_{j+1}} = 0$  for some  $k_j$ , and therefore the products are both zero in Eq. (3).

These results can be put to good use. In particular, note that the probability  $P(S_i = 1)$  is equal to the sum of the probabilities of all paths containing state  $s_i$ . Further, since  $s_i$  has  $q - 1$  processes in its completed set,  $s_i$  will be the  $q$ th state reached in any path which contains  $s_i$ . Therefore, if the sequence  $(k_1, k_2, \dots, k_{n+1})$  is

identical to some path  $\alpha$  containing  $s_i$ , we must have  $k_q = \alpha_q = i$ . Of course, we must also have  $k_1 = 1$  and  $k_{n+1} = f$ . It follows that

$$P(S_i = 1) = \sum_{k_2=1}^f \sum_{k_3=1}^f \cdots \sum_{k_{q-1}=1}^f \sum_{k_{q+1}=1}^f \cdots \sum_{k_n=1}^f [d_{k_1 k_2} d_{k_2 k_3} \cdots d_{k_{q-1} k_q} d_{k_q k_{q+1}} \cdots d_{k_n k_{n+1}}].$$

Note that it would be superfluous to sum over  $k_1$  or  $k_{n+1}$  since any path must begin with the start state  $s_{k_1} = s_1$  and end with the finish state  $s_{k_{n+1}} = s_f$ . Rearranging the above equation, one obtains

$$P(S_{k_q} = 1) = \sum_{k_2=1}^f \cdots \sum_{k_{q-1}=1}^f \left\{ d_{k_1 k_2} \cdots d_{k_{q-1} k_q} \times \left[ \sum_{k_{q+1}=1}^f d_{k_q k_{q+1}} \sum_{k_{q+2}=1}^f d_{k_{q+1} k_{q+2}} \cdots \sum_{k_n=1}^f d_{k_{n-1} k_n} d_{k_n k_{n+1}} \right] \right\}.$$

Since  $d_{k_n k_{n+1}} = 1$  ( $s_{k_n}$  has only one task in its current set) and since  $\sum_{k_j=1}^f d_{k_{j-1} k_j} = 1$ , the expression in square brackets reduces to 1. Therefore,

$$P(S_{k_q} = 1) = \sum_{k_2=1}^f \cdots \sum_{k_{q-1}=1}^f d_{k_1 k_2} d_{k_2 k_3} \cdots d_{k_{q-1} k_q}.$$

It is not hard to show that this expression is equal to the  $(k_1, k_q)$  entry of the matrix  $\mathbf{D}^{q-1}$ . However,  $k_1 = 1$ . Thus,

$$P(S_{k_q} = 1) = \mathbf{e}'_1 \mathbf{D}^{q-1} \mathbf{e}_{k_q}.$$

Since by assumption  $k_q = i$ , this completes the proof of Lemma 11.

**LEMMA 12.** *Suppose state  $s_i$  has  $q - 1$  processes in its completed set and that state  $s_j$  has  $r - 1$  processes in its completed set, and assume that  $r > q$ . Then,*

$$P(S_i = 1, S_j = 1) = [\mathbf{e}'_i \mathbf{D}^{q-1} \mathbf{e}_i][\mathbf{e}'_j \mathbf{D}^{r-q} \mathbf{e}_j].$$

*Proof.* Following the logic of Lemma 11, one can write

$$\begin{aligned} P(S_i = 1, S_j = 1) &= \sum_{k_2=1}^f \cdots \sum_{k_{q-1}=1}^f \sum_{k_{q+1}=1}^f \cdots \sum_{k_{r-1}=1}^f \sum_{k_{r+1}=1}^f \cdots \sum_{k_n=1}^f \\ &\quad [d_{k_1 k_2} \cdots d_{k_{q-1} k_q} d_{k_q k_{q+1}} \cdots d_{k_{r-1} k_r} d_{k_r k_{r+1}} \cdots d_{k_n k_{n+1}}] \\ &= \sum_{k_2=1}^f \cdots \sum_{k_{q-1}=1}^f d_{k_1 k_2} \cdots d_{k_{q-1} k_q} \sum_{k_{q+1}=1}^f \cdots \sum_{k_{r-1}=1}^f d_{k_q k_{q+1}} \cdots d_{k_{r-1} k_r} \\ &\quad \times \left[ \sum_{k_{r+1}=1}^f \cdots \sum_{k_n=1}^f d_{k_r k_{r+1}} \cdots d_{k_n k_{n+1}} \right] \\ &= \sum_{k_2=1}^f \cdots \sum_{k_{q-1}=1}^f d_{k_1 k_2} \cdots d_{k_{q-1} k_q} \sum_{k_{q+1}=1}^f \cdots \sum_{k_{r-1}=1}^f d_{k_q k_{q+1}} \cdots d_{k_{r-1} k_r} \end{aligned}$$

since, as before, the expression in brackets reduces to 1. Thus

$$P(S_{k_q} = 1, S_{k_r} = 1) = [\mathbf{e}'_1 \mathbf{D}^{q-1} \mathbf{e}_{k_q}] [\mathbf{e}'_{k_q} \mathbf{D}^{r-q} \mathbf{e}_{k_r}],$$

which, rewriting, becomes the first expression. This completes the proof of Lemma 12.

Note that if  $i \neq j$  but  $q = r$ , then  $P(S_i = 1, S_j = 1) = 0$ . That is, two distinct states with the same number of processes in their completed sets cannot both be reached on a single trial. This is consistent with Lemma 12 since  $\mathbf{D}^0 = \mathbf{I}$  and since the expression on the right will equal 0. Also note that if  $i = j$  (and therefore  $q = r$ ), the term on the right will equal 1, and the expression reduces to that for  $P(S_i = 1)$ .

**THEOREM 3.** *Let  $r_i$  denote the number of processes in the completed set of state  $s_i$ . Then*

$$E[T] = \sum_{i=1}^{f-1} (1/c_i) \mathbf{e}'_1 \mathbf{D}^{r_i} \mathbf{e}_i.$$

*Proof.* Substitute from Corollary 3 and Lemma 11 into Eq. (1).

**THEOREM 4.** *Let  $r_i$  denote the number of processes in the completed set of state  $s_i$ . Assume that states are labelled so that  $j > i$  implies  $r_j \geq r_i$ . Then,*

$$\begin{aligned} \text{VAR}[T] = & \sum_{i=1}^{f-1} \{ (1/c_i)^2 [\mathbf{e}'_1 \mathbf{D}^{r_i} \mathbf{e}_i] [2 - \mathbf{e}'_1 \mathbf{D}^{r_i} \mathbf{e}_i] \} \\ & + 2 \sum_{i < j} \{ [1/(c_i c_j)] [\mathbf{e}'_1 \mathbf{D}^{r_i} \mathbf{e}_i] [(\mathbf{e}'_i \mathbf{D}^{r_j - r_i} \mathbf{e}_j) - (\mathbf{e}'_i \mathbf{D}^{r_i} \mathbf{e}_j)] \}. \end{aligned}$$

*Proof.* Substitute from Corollaries 3, 4 and Lemmas 11, 12 into Eq. (2) and simplify.

### Example Computations: The Mean

It may be useful to illustrate the required calculations for several example PERT networks. First, consider computations of the expected task duration of the PERT network displayed in Fig. 5a. (Note that the expected task duration in this situation is simply the expectation of the maximum of two independent, exponentially distributed random variables.) The associated OP diagram is displayed in Fig. 5b. Theorem 3 stated immediately above gives the required calculations for the expected task duration. Since some readers may have skipped the proofs and explanation of the notation in Theorem 3, a brief explanation is in order.

Six quantities require interpretation: the constants  $f$ ,  $c_i$ , and  $r_i$ , the row vector  $\mathbf{e}'_1$ , the column vector  $\mathbf{e}_i$ , and the matrix  $\mathbf{D}$ . The constant  $f$  is equal to the total number of states in the OP diagram ( $f=4$  in the example in Fig. 5). The last state to be executed must be labelled  $s_f$  and the first state to be executed must be labelled  $s_1$ . In

labelling the intermediate states, it is convenient to follow the convention that a state with a larger number of processes in its completed set be given a larger index than a state with fewer processes in its completed set. It is always possible to label the states in accordance with this convention. Only states  $s_1$  through  $s_{f-1}$  are used in the computation of the expected task duration.

The quantity  $c_i$  is equal to the sum of the rate parameters of the processes current in state  $s_i$ . For example, in state  $s_1$  two processes are current,  $x_1$  and  $x_2$  (see Fig. 5) with rate parameters respectively of  $\lambda_1$  and  $\lambda_2$ . (Recall that it is assumed that the duration  $X_i$  of each process  $x_i$  is exponentially distributed with rate parameter  $\lambda_i$ , i.e., the density function  $f(x_i)$  is defined as  $\lambda_i \exp[-\lambda_i x_i]$ .) Thus, in the current example,  $c_1 = \lambda_1 + \lambda_2$ ,  $c_2 = \lambda_2$ , and  $c_3 = \lambda_3$ .

The matrix  $\mathbf{D}$  is a square  $f \times f$  matrix. The component  $d_{ij}$  in the  $i$ th row and the  $j$ th column is equal to zero if state  $s_j$  does not follow immediately after state  $s_i$  on a path in the OP diagram. So, for example,  $d_{14} = 0$  since state  $s_4$  does not follow state  $s_1$  immediately. The component  $d_{ij}$  is positive if state  $s_j$  does follow immediately after state  $s_i$  on a path in the OP diagram. If such is the case, then  $d_{ij}$  is set equal to the quotient formed by placing the rate parameter of the process that moves the system from state  $s_i$  to state  $s_j$  in the numerator and by placing the quantity  $c_i$  in the denominator. For example,  $d_{12} = \lambda_1 / (\lambda_1 + \lambda_2)$  since the completion of process  $x_1$  moves the system from state  $s_1$  to state  $s_2$ . Similarly,  $d_{13} = \lambda_2 / (\lambda_1 + \lambda_2)$  since the completion of process  $x_2$  moves the system from state  $s_1$  to state  $s_3$ . The only other positive entries in  $\mathbf{D}$  are  $d_{24}$  and  $d_{34}$ :  $d_{24} = \lambda_2 / \lambda_2 = 1$  and  $d_{34} = \lambda_1 / \lambda_1 = 1$ . All other  $d_{ij}$  are set equal to zero. Recall that  $\mathbf{D}^0 = \mathbf{I}$ , where  $\mathbf{I}$  is the identity matrix, i.e., a matrix with 1's on the diagonal and 0's everywhere else.

The quantity  $r_i$  is set equal to the number of processes in the completed set of state  $s_i$ . Thus, since there are no processes in the completed set of state  $s_1$ , we have  $r_1 = 0$ . Similarly,  $r_2 = r_3 = 1$  and  $r_4 = 2$ .

Finally, the row vector  $\mathbf{e}'_i$  of dimension  $f$  has a 1 in column  $i$  and 0's everywhere else. The column vector  $\mathbf{e}_j$  is also of dimension  $f$  and has a 1 in row  $j$  and 0's everywhere else.

Putting all of this information together and using Theorem 3, one obtains

$$\begin{aligned} E[T] &= \sum_{i=1}^3 (1/c_i) \mathbf{e}'_i \mathbf{D}^{r_i} \mathbf{e}_i \\ &= (1/c_1) \mathbf{e}'_1 \mathbf{D}^{r_1} \mathbf{e}_1 + (1/c_2) \mathbf{e}'_1 \mathbf{D}^{r_2} \mathbf{e}_2 + (1/c_3) \mathbf{e}'_1 \mathbf{D}^{r_3} \mathbf{e}_3 \\ &= [1/(\lambda_1 + \lambda_2)] \mathbf{e}'_1 \mathbf{D}^0 \mathbf{e}_1 + (1/\lambda_2) \mathbf{e}'_1 \mathbf{D}^1 \mathbf{e}_2 + (1/\lambda_1) \mathbf{e}'_1 \mathbf{D}^1 \mathbf{e}_3 \\ &= [1/(\lambda_1 + \lambda_2)] [1] + [1/\lambda_2] [\lambda_1 / (\lambda_1 + \lambda_2)] + [1/\lambda_1] [\lambda_2 / (\lambda_1 + \lambda_2)]. \end{aligned}$$

One can easily show by other methods that the above quantity is indeed the expected value of the maximum of two independent, exponentially distributed random variables (e.g., see Townsend, 1974, p. 149, Eq. 6). This completes the discussion of the computations required to compute the expected task duration of the PERT network in Fig. 5.

The computations for more complicated networks require the specification of exactly the same sort of information. Thus, for each state  $s_i$  in the OP diagram of the relevant PERT network, the quantities  $c_i$ ,  $\mathbf{D}$ , and  $r_i$  must be given ( $\mathbf{e}'_i$  and  $\mathbf{e}_i$  follow by definition). For the OP diagram as a whole, the number of states  $f$  must also be known. Briefly, consider the OP diagram in Fig. 6. There are a total of 12 states in the diagram (so  $f = 12$ ). Consider state  $s_3$ . Since processes  $x_2, x_3$ , and  $x_4$  are current,  $c_3 = \lambda_2 + \lambda_3 + \lambda_4$ . Since process  $x_4$  completes if a transition is made from state  $s_3$  to state  $s_4$ ,  $d_{34} = \lambda_4 / (\lambda_2 + \lambda_3 + \lambda_4)$ . Similarly,  $d_{35} = \lambda_3 / (\lambda_2 + \lambda_3 + \lambda_4)$  and  $d_{36} = \lambda_2 / (\lambda_2 + \lambda_3 + \lambda_4)$ . Finally,  $r_3 = 1$  since there is one process in the completed set of state  $s_3$ . One proceeds through the remaining states in exactly the same fashion after which Theorem 3 can be used to compute the expected task duration.

*The Memoryless Property*

Before concluding this section, a final remark about an initially attractive alternative derivation of several of the above lemmas and theorems is in order. It can easily be shown that if a random variable  $X_i$  has an exponential distribution then  $P(X_i - h > t | X_i > h) = P(X_i > t)$ . One might ask why this “memoryless” property of exponential random variables cannot be used to determine, say, the expected task duration of a state in the OP diagram, given that the state is entered along path  $\alpha$  and given that there is only one process in the current set. In light of the above discussion, the relevant conditional probability can be written as

$$\begin{aligned}
 &P(T_{\alpha_i} > t | T_{\alpha_i} > 0, T_{\alpha_{i-1}} > 0, \dots, T_{\alpha_1} > 0) \\
 &= P \left( X_{\alpha_i}^* - \sum_{j=1}^{i-1} a_{ij} T_{\alpha_j} > t \mid \right. \\
 &\quad 0 < X_{\alpha_{i-1}}^* - \sum_{j=1}^{i-2} a_{i-1,j} T_{\alpha_j} < \left. \left\{ X_{\alpha_k}^* - \sum_{j=1}^{i-2} a_{kj} T_{\alpha_j} \mid \alpha_k^* \in C_{\alpha_{i-1}} - \{\alpha_{i-1}^*\} \right\}, \right. \\
 &\quad 0 < X_{\alpha_{i-2}}^* - \sum_{j=1}^{i-3} a_{i-2,j} T_{\alpha_j} < \left. \left\{ X_{\alpha_k}^* - \sum_{j=1}^{i-3} a_{kj} T_{\alpha_j} \mid \alpha_k^* \in C_{\alpha_{i-2}} - \{\alpha_{i-2}^*\} \right\}, \right. \\
 &\quad \vdots \\
 &\quad \left. 0 < X_{\alpha_1}^* < \left\{ X_{\alpha_k}^* \mid \alpha_k^* \in C_{\alpha_1} - \{\alpha_1^*\} \right\} \right).
 \end{aligned}$$

Unfortunately, the memoryless property cannot be applied as is to the above equation since the memoryless property is stated in terms of the constants “ $t$ ” and “ $h$ ” and not in terms of a constant “ $t$ ” and a linear combination of random variables. Moreover, the event on which the above probability is conditioned is much more complicated than the one involved in the definition of the memoryless property.<sup>1</sup>

<sup>1</sup> Both reviewers of this paper and the Editor suggested that the standard (univariate) memoryless property could be used to shorten the proofs. The authors made several attempts to shorten the proofs in the direction suggested, but were not successful.

### Distribution Function

The density function  $f_T(t)$  of the task duration  $T$  can be written as the sum over all paths in the OP diagram of the product of the conditional density  $f_{T|P_\alpha=1}(t)$  times the probability that path  $\alpha$  is taken, i.e.,

$$f_T(t) = \sum_{\alpha \in \Gamma} f_{T|P_\alpha=1}(t) P(P_\alpha = 1). \quad (4)$$

By Corollary 2 the conditional density is the convolution of a set of mutually independent, exponentially distributed random variables each with rate parameter  $c_{\alpha_i}$ .

McGill and Gibbon (1965) show how to obtain the conditional density. Briefly, they define the time constant  $w_{in}$  as

$$w_{in} = \left[ \prod_{j \neq i} (1 - c_{\alpha_i}/c_{\alpha_j}) \right]^{-1},$$

where the notation  $\prod_{j \neq i}$  indicates the product of the terms where  $j$  takes on values from 1 to  $n$ , except for  $j=i$ . Then the conditional density function of the task duration can be written quite simply as a weighted sum of exponentials

$$f_{T|P_\alpha=1}(t) = \sum_{i=1}^n w_{in} c_{\alpha_i} \exp[-c_{\alpha_i} t] \quad \text{for } t \geq 0.$$

These observations lead to

THEOREM 5.

$$f_T(t) = \sum_{\alpha \in \Gamma} \left\{ \left[ \sum_{i=1}^n w_{in} c_{\alpha_i} \exp(-c_{\alpha_i} t) \right] \left[ \prod_{i=1}^n \lambda_{\alpha_i}/c_{\alpha_i} \right] \right\}.$$

*Proof.* Substitute the expression for  $f_{T|P_\alpha=1}(t)$  and the expression for  $P(P_\alpha = 1)$  from Theorem 2 into Eq. (4). The distribution function is then obtained by integrating over  $t$ .

### DISTRIBUTION OF TASK DURATION: GENERAL-GAMMA CASE

The above methods have been discussed in the context of models which assume that the constituent process durations are mutually independent, exponentially distributed random variables. It should be noted that the above methods can also be used to determine closed form expressions for the mean, variance, and distribution of the task duration when the durations of individual processes can be expressed as the *sum* of two or more mutually independent random variables, each of which is itself exponentially distributed (note that this includes as a special case a  $\text{gamma}(k, \beta)$  when the  $k$  exponential random variables are independent and identically distributed with common rate parameter  $\beta$ ). To see this, note that each process can be considered as a set of subprocesses where each subprocess has a completion time which is exponentially distributed.

For example, consider the PERT network in Fig. 7a. This network defines a



parallel system with two processes,  $x_1$  and  $x_2$ . Suppose that the distribution of the duration of each process can be expressed as the sum of two independent, exponential random variables. Let  $x_{11}$  and  $x_{12}$  be the subprocesses associated with process  $x_1$  and let  $\lambda_{11}$  and  $\lambda_{12}$  be the corresponding rate parameters. Similarly, let  $x_{21}$  and  $x_{22}$  be the subprocesses associated with process  $x_2$  and let  $\lambda_{21}$  and  $\lambda_{22}$  be the corresponding rate parameters. Then the PERT network in Fig. 7a can be expressed as the augmented PERT network in Fig. 7b. The construction of the OP diagram and the computation of moments of the distribution of the task proceed as usual.

This line of thought can be extended further. Rather than replace a process with a set of subprocesses in series, we could replace a process with an entire PERT network of subprocesses (cf. Hartley and Wortham, 1966; Ringer, 1969). Thus, the process durations need not be restricted to general-gamma; certain probability mixtures of general-gamma distributions can serve as well. However, this level of generality will not be pursued any further in this paper.

#### DISCUSSION

In the first part of this section, we illustrate the Order-of-Processing method with applications to the study of two psychological phenomena, memory and visual search. In the second and final part of this section, the flexibility of the Order-of-Processing method is emphasized.

##### *Memory and Visual Search*

The study of memory and visual search has been the focus of continuing attention

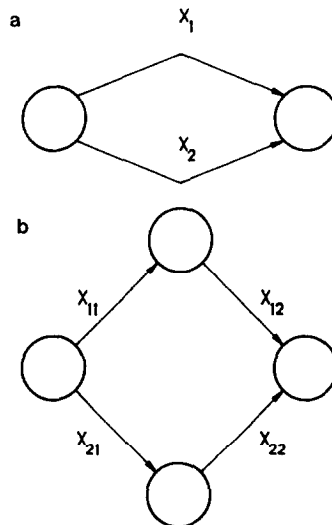


FIG. 7. (a) A two-process parallel PERT network and (b) an example augmentation to cover situations in which the process durations are mutually independent, general-gamma distributed random variables.

for the past twenty years or so. (For a recent review, see Schneider and Shiffrin, 1977; Shiffrin and Schneider, 1977.) In broad outline, many memory and visual search tasks can be viewed as the problem of identifying a target character in an array of characters. For example, in a standard memory scanning task (e.g., Sternberg, 1966), subjects must indicate on each trial whether a single target display character appears in a previously memorized set of stimuli. In a typical visual search experiment, subjects are told to scan a display of characters for a prespecified target. The target appears on a random one-half of the trials; only distractors appear on the remaining trials.

In both the memory scanning task and the visual search task, the core of the problem for the subject is to execute a set of cognitive processes each of which results in the comparison between an encoded representation of the target character and an encoded representation of one of the characters in the search set. Of course, the subject must also execute the cognitive processes which result in the characters being encoded. However, in the memory scanning task this is accomplished for all stimuli in the search set prior to the onset of the trial; only the target character remains to be encoded. Thus, it will simplify matters if, for the moment, we restrict attention to the memory scanning task.

A number of different models of the memory scanning task have been developed (e.g., see Sternberg, 1975). The simplest models differ in comparison mechanisms (serial or parallel) and search strategies (self-terminating or exhaustive). A serial comparison mechanism is one which executes only one comparison at a time, while a parallel comparison mechanism is one which executes all comparisons simultaneously. A search strategy is self-terminating if comparison activity stops as soon as the target is identified, while a search strategy is exhaustive if comparison activity does not stop until all characters in the search set have been examined.

The Order-of-Processing method can be used to formulate and quantify models with the above mechanisms and strategies. Immediately below, application of the OP method to both serial and parallel comparison models of memory scanning are discussed. Consideration is given only to the exhaustive search strategy. Application of the method to models with a self-terminating search strategy would require a more extensive discussion of task systems than is warranted in the current context.

In the discussion that follows, it will be assumed that there are  $n$  stimuli in the (encoded) search set. Let  $x_1, \dots, x_n$  denote the corresponding comparison processes. That is,  $x_i$  denotes the comparison of the target character with the  $i$ th stimulus in the search set. (An alternate formulation would be to define  $x_i$  as a process which includes the random selection of a stimulus from the search set, but this seems an unnecessary complication for our present purposes of illustration.)

*Serial Comparison.* Consider first a serial comparison model. In particular, consider a serial comparison model with exhaustive search. Thus, each of the processes  $x_1, \dots, x_n$  must be executed in turn. For the sake of simplicity, let  $x_0$  be the only process which precedes the first comparison and let  $x_{n+1}$  be the only process which follows the last comparison. We intend the process  $x_0$  to include the encoding

of the target stimulus and the process  $x_{n+1}$  to include whatever motor processes are involved in responding. Assume that the durations of all processes are mutually independent, exponentially distributed random variables.

The exhaustive search version of the serial comparison model is described by a PERT network having exactly one path, along which appear the processes  $x_0, x_1, \dots, x_{n+1}$  in that order. Thus, the OP diagram describing the model will consist of a single path having  $n + 3$  states. For  $i = 0, 1, \dots, n + 1$ , the current set of state  $s_i$  consists of the process  $x_i$ . Of course, the current set of the final state  $s_{n+2}$  is empty. For  $i = 1, 2, \dots, n + 2$ , the completed set of state  $s_i$  is  $\{x_0, x_1, \dots, x_{i-1}\}$ . As usual, the completed set of the initial state  $s_0$  is empty. The expectation of the task duration  $T$  of a serial exhaustive search is the sum of the expectations of the state durations  $T_j$  ( $0 \leq j \leq n + 1$ ). The computations follow directly from the discussion in the previous section. In this simple serial case, no computational advantage is achieved by using the OP method over more traditional methods.

*Parallel Comparison.* Consider next a parallel comparison model of a memory search task. If it is assumed that process  $x_0$  must complete before processes  $x_1, x_2, \dots, x_n$  begin, that processes  $x_1, x_2, \dots, x_n$  are executed in parallel, and that all processes  $x_j$  ( $j \leq n$ ) must finish before process  $x_{n+1}$  begins, then the OP diagram would be constructed as follows. There is one state, say state  $s_0$ , at level 0 and one process,  $x_0$ , in its current set. There is also one state, say state  $s_1$ , at level 1,  $n$  processes ( $x_1, \dots, x_n$ ) in the current set of this state and one process,  $x_0$ , in its completed set. At each level  $i$  ( $1 \leq i \leq n$ ) there will be a total of  $\binom{n}{i-1}$  states. Each state will have  $i$  processes in the completed set and  $n - i + 1$  processes in the current set. Formation of the contents of the current and completed sets of states at each level  $i$  ( $1 \leq i \leq n$ ) follows in a straightforward fashion from the discussion of the construction of OP diagrams in an earlier section. There will be one process,  $x_{n+1}$ , in the current set of the one state at level  $n + 1$  and there will be  $n + 1$  processes ( $x_0, x_1, \dots, x_n$ ) in the completed set. Finally, all  $n + 2$  processes will appear in the completed set of the one state at level  $n + 2$ . Computation of the expected duration of a parallel exhaustive search can be computed using the results presented in the previous section. While the OP method did not facilitate computation of the desired quantities for the serial model, it does systematize the computations required for the parallel model.

It can be noted in passing that the OP diagram makes transparent the reason parallel systems with constant processing rates are not identifiably different from serial systems with different processing orders (see, e.g., Townsend, 1972). Consider the simple parallel PERT network at the top of Fig. 5. Assume that  $2\lambda_1 = \lambda_2$ . Then an equivalent serial system is (a) one in which  $x_1$  is processed before  $x_2$  with probability  $\lambda_1/(\lambda_1 + \lambda_2) = \frac{1}{3}$  (the probability in the parallel system of making a transition from state  $s_1$  to state  $s_2$ ), in which the rate parameter governing the duration of  $x_1$  is  $\lambda_1 + \lambda_2$ , and in which the rate parameter governing the duration of  $x_2$  is  $2\lambda_1$ ; and (b) one in which  $x_2$  is processed before  $x_1$  with probability  $\frac{2}{3}$  (the probability in the parallel system of making a transition from state  $s_1$  to state  $s_3$ ), in which the rate

parameter governing the duration of  $x_2$  is  $\lambda_1 + \lambda_2$ , and in which the rate parameter governing the duration of  $x_1$  is  $\lambda_1$ .

*Serial-Parallel Comparison.* Consider next a model of visual search behavior. As mentioned earlier, subjects in a typical visual search experiment are told to scan a display of characters for a prespecified target. The target appears on a random one-half of the trials; only distractors appear on the remaining trials. Thus, in contrast to the memory scanning task, the target stimulus is encoded prior to the onset of the trial. However, the stimuli in the search set must be encoded during the trial.

One possible hybrid serial-parallel model of visual search has been described by Fisher (Note 3). It is assumed that the characters in the display are encoded one at a time. It is also assumed that the number of comparison processes which can occur simultaneously is limited and greater than one. That is, it is assumed that the subject can compare up to  $k$  display characters with the target at any one moment in time.

A PERT network can be used to represent the serial-parallel comparison model when the number of stimuli in the search set does not exceed  $k$ . For example, suppose there are two stimuli in the search set and  $k = 2$  (see Fig. 8a). Process  $E_1$  refers to the encoding or scanning of the first stimulus to which attention is directed, while process  $E_2$  refers to the encoding or scanning of the second stimulus to which attention is directed. Process  $C_3$  refers to the comparison of the stimulus scanned by process  $E_1$  with the target, while process  $C_4$  refers to the comparison of the stimulus scanned by process  $E_2$  with the target. This somewhat awkward notation for the labelling of the processes was chosen because it considerably simplifies the labelling of the corresponding OP diagram. Note that the scanning processes  $E_1$  and  $E_2$  are executed sequentially whereas the comparison processes  $C_3$  and  $C_4$  are executed concurrently. The response process  $R_5$  cannot begin until both comparison process  $C_3$  and  $C_4$  are finished. Since the case of  $k = 2$  with two search stimuli is representable as a PERT network, there is no need to comment here on the construction of the corresponding OP diagram (see, Fig. 8b). Computation of the expected serial-parallel exhaustive task duration proceeds as usual.

#### *Flexibility of the OP Method*

The above approach is useful if the durations of the individual processes in the system are mutually independent, exponentially distributed random variables. One might well ask whether the durations of psychological processes or tasks are well approximated by exponential random variables. Some recent evidence suggests that this is indeed the case (Ashby, 1982; Ashby and Townsend, 1980; Kofeld, Santee, and Wallace, 1981), at least for certain mental processes. However, it is important to bear in mind that the above approach can handle more complex distributions. Specifically, it was noted that the approach can be used to model the durations of processes which have a gamma distribution, and more generally, the durations of tasks which are *sums* of mutually independent, exponentially distributed random variables. These distributions do not have the seemingly counterintuitive memoryless property of the exponential distribution. In addition, with an appropriate choice of the

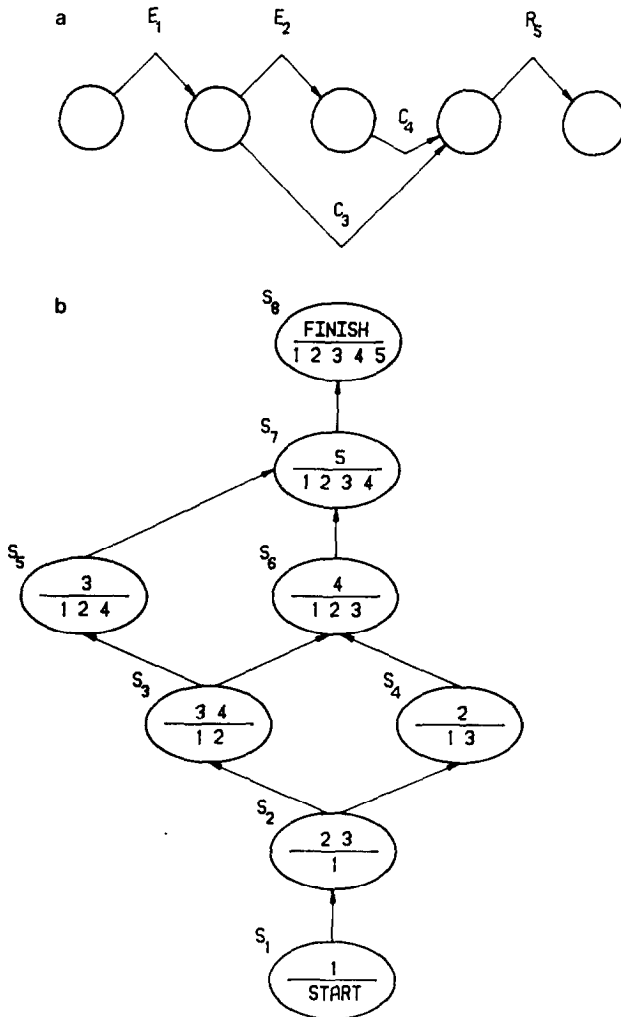


FIG. 8. (a) A PERT network used to represent the processing in a serial-parallel comparison model when there are two stimuli in the search set and two comparison processors. (b) The associated OP diagram.

parameters these distributions are single peaked and have a skewness to the right, two properties which are found in many empirical studies of the reaction time distribution.

CONCLUSION

Psychological behaviors are presumably mediated by one or more cognitive processes. Reaction time is a function of the arrangement of the processes and of the distributions of the process durations. It was shown how to compute closed form

expressions for the mean, variance, and distribution of reaction time when the processes can be arranged in a PERT network and when the durations of the processes are either mutually independent exponential random variables or sums of mutually independent exponential random variables. The method relied heavily on the translation of the partial ordering of the processes in a PERT network into an ordering on the processes in an Order-Of-Processing (OP) diagram. It was also shown that the OP diagram could be used to obtain the expected exhaustive-search task durations of serial, parallel and hybrid serial-parallel models of memory and visual search.

Schweickert (1978, 1980) has shown how to determine the arrangement of processes in a PERT network when it is assumed that the durations of the individual processes are constants. It remains to be seen whether the above methods can be used to determine the arrangement of the processes when the durations of the processes vary from trial to trial.

#### REFERENCES

- ASHBY, F. G. Testing the assumption of exponential additive reaction time distributions. *Memory and Cognition*, 1982, **10**, 125–134.
- ASHBY, F. G., & TOWNSEND, J. T. Decomposing the reaction time distribution: pure insertion and selective influence revisited. *Journal of Mathematical Psychology*, 1980, **21**, 93–123.
- ATKINSON, R. C., HOLMGREN, J. E., & JUOLA, J. F. Processing time as influenced by the number of elements in a visual display. *Perception and Psychophysics*, 1969, **6**, 321–326.
- COFFMAN, E. G. (Ed.) Introduction to deterministic scheduling theory. In *Computer and job-shop scheduling theory*. New York: Wiley, 1976.
- CHRISTIE, L. S., & LUCE, R. D. Decision structure and time relations in simple choice behavior. *Bulletin of Mathematical Biophysics*, 1956, **18**, 89–112.
- DONDERS, F. C. Die Schnelligkeit psychischer Prozesse. *Archive fur Anatomie und wissenschaftliche Medizen*, 1868, **656**–681.
- EGETH, H., JONIDES, J., & WALL, S. Parallel processing of multielement displays. *Cognitive Psychology*, 1972, **3**, 674–698.
- FISHER, D. L. A three-factor model of syllogistic reasoning: The study of isolable stages. *Memory and Cognition*, 1981, **9**, 496–514.
- FISHER, D. L. Limited channel models of automatic detection: Capacity and scanning in visual search. *Psychological Review*, 1982, **89**, 662–692.
- FULKERSON, D. R. Expected critical path length in PERT networks. *Operations Research*, 1962, **10**, 808–817.
- HARTLEY, H. O., & WORTHAM, A. W. A statistical theory for PERT critical path analysis. *Management Science*, 1966, **12**, B-469–B-481.
- KLEINDORFER, G. B. Bounding distributions for a stochastic acyclic network. *Operations Research*, 1971, **19**, 1586–1601.
- KOFELD, D. L., SANTEE, J. L., & WALLACE. Loudness and reaction time. II. Identification of detection components at different intensities and frequencies. *Perception and Psychophysics*, 1981, **29**, 550–562.
- MARTIN, J. J. Distribution of the time through a directed acyclic network. *Operations Research*, 1965, **13**, 46–66.
- MCGILL, W. J. Stochastic latency mechanisms. In R. D. Luce, R. R. Bush, and E. Galanter (Eds.), *Handbook of Mathematical Psychology*, Vol. 1. New York: Wiley, 1963.
- MCGILL, W. J., & GIBBON, J. The general-gamma distribution and reaction times. *Journal of Mathematical Psychology*, 1965, **2**, 1–18.

- MODOR, J. J., & PHILLIPS, C. R. *Project management with CPM and PERT* (2nd ed.). New York: Van Nostrand-Reinhold, 1970.
- NADAS, A. Probabilistic PERT. *IBM Journal of Research and Development*, 1979, **23**, 339-347.
- PRITSKER, A. A. B., & HAPP, W. W. GERT: Graphical evaluation and review technique. Part I. Fundamentals. *The Journal of Industrial Engineering*, 1966, **17**, 266-267.
- PRITSKER, A. A. B., & WHITEHOUSE, G. E. GERT: Graphical evaluation and review technique. Part II. Probabilistic and industrial engineering applications. *The Journal of Industrial Engineering*, 1966, **17**, 293-301.
- RINGER, L. R. Numerical operators for statistical PERT critical path analysis. *Management Science*, 1969, **16**, B-136-B-143.
- ROBILLARD, P., & TRAHAN, M. The completion time of PERT networks. *Operations Research*, 1977, **25**, 15-29.
- SCHNEIDER, W., & SHIFFRIN, R. M. Controlled and automatic human information processing. I. Detection, search, and attention. *Psychological Review*, 1977, **84**, 1-66.
- SHIFFRIN, R. M., & SCHNEIDER, W. Controlled and automatic human information processing. II. Perceptual learning, automatic attending, and a general theory. *Psychological Review*, 1977, **84**, 127-190.
- SCHWEICKERT, R. A critical path generalization of the additive factor method: Analysis of a Stroop task. *Journal of Mathematical Psychology*, 1978, **18**, 105-139.
- SCHWEICKERT, R. Critical-path scheduling of mental processes in a dual task. *Science*, 1980, **209**, 704-706.
- STERNBERG, S. High-speed scanning in human memory. *Science*, 1966, **153**, 652-654.
- STERNBERG, S. Scanning a persisting visual image versus a memorized list. Paper presented at the meeting of the Eastern Psychological Association, Boston, April, 1967.
- STERNBERG, S. The discovery of processing stages: Extensions of Donder's method. In W. G. Koster (Ed.), *Attention and Performance II*. Amsterdam: North-Holland, 1969.
- STERNBERG, S. Memory scanning: New findings and current controversies. *Quarterly Journal of Experimental Psychology*, 1975, **27**, 1-32.
- TAYLOR, D. A. Stage analysis of reaction time. *Psychological Bulletin*, 1976, **83**, 161-191.
- TOWNSEND, J. T. A note on the identifiability of parallel and serial processors. *Perception and Psychophysics*, 1971, **10**, 161-173.
- TOWNSEND, J. T. Some results concerning the identifiability of parallel and serial processes. *British Journal of Mathematical and Statistical Psychology*, 1972, **25**, 168-199.
- TOWNSEND, J. T. Issues and models concerning the processing of a finite number of inputs. In B. H. Kantowitz (Ed.), *Human information processing: Tutorials in performance and cognition*. Potomac, Md.: Erlbaum, 1974.

## REFERENCE NOTES

1. FISHER, D. L., & GOLDSTEIN, W. M. *Complex cognitive tasks: Probabilistic PERT and models of reaction time*. Paper presented at the fourteenth annual meetings of the Society for Mathematical Psychology, Santa Barbara, August 1981.
2. FISHER, D. L., & GOLDSTEIN, W. M. *Probabilistic PERT: A continuous time Markov process*. Paper presented at the meetings of the Operations Research Society of America, Detroit, April 1982.
3. FISHER, D. L. *Stochastic models of visual search: queues, PERT networks, and OP diagrams*. Unpublished doctoral dissertation, University of Michigan, 1983.

RECEIVED: May 20, 1981