

UNCLASSIFIED

AD 274 597

*Reproduced
by the*

**ARMED SERVICES TECHNICAL INFORMATION AGENCY
ARLINGTON HALL STATION
ARLINGTON 12, VIRGINIA**



UNCLASSIFIED

NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

74597

274 597

MEMORANDUM

RM-3075-PR

MARCH 1962

CATALOGED BY AD117
AS AD NO. _____

EXPECTED CRITICAL PATH LENGTHS IN PERT NETWORKS

D. R. Fulkerson

PREPARED FOR:

UNITED STATES AIR FORCE PROJECT RAND

ASTIA
RECEIVED
APR 27 1962
ASTIA

739200

The **RAND** Corporation
SANTA MONICA • CALIFORNIA

MEMORANDUM

RM-3075-PR

MARCH 1962

**EXPECTED CRITICAL PATH LENGTHS
IN PERT NETWORKS**

D. R. Fulkerson

This research is sponsored by the United States Air Force under Project RAND — Contract No. AF 49(638)-700 — monitored by the Directorate of Development Planning, Deputy Chief of Staff, Research and Technology, Hq USAF. Views or conclusions contained in this Memorandum should not be interpreted as representing the official opinion or policy of the United States Air Force. Permission to quote from or reproduce portions of this Memorandum must be obtained from The RAND Corporation.

The **RAND** *Corporation*

1700 MAIN ST • SANTA MONICA • CALIFORNIA

PREFACE

The calculation of project duration times and project costs by means of network models has become increasingly popular within the last few years. These models, which go by such names as PERT (Program Evaluation Review Technique), PEP (Program Evaluation Procedure), Critical Path Scheduling, Project Cost Curve Scheduling, and others, have the common feature that uncertainties in job times are either ignored or handled outside the network analysis, usually by replacing each distribution of job times by its expected value.

This Project RAND Memorandum describes a feasible computational method that seems to yield a fairly good approximation to the expected duration time of a project whose individual job times are random variables.

Another possible area of application of this method would be to communication networks whose components are subject to random delay times.

SUMMARY

A practicable method of computing an approximation to the expected critical path length in a PERT network in which arc lengths are random variables is proposed. The usual practice in this situation is simply to replace each distribution by its expected value, thereby obtaining a deterministic problem whose solution provides an estimate g of the expected critical path length e . This estimate is always optimistic: $g \leq e$. The proposed method yields a usually better, and never worse, approximation f satisfying $g \leq f \leq e$.

CONTENTS

PREFACE111
SUMMARY	v
Section	
1. INTRODUCTION.	1
2. PERT NETWORKS	2
3. ASSUMPTIONS AND NOTATION.	3
4. THE NUMBERS f_1	7
5. SOME NUMERICAL EXAMPLES	13
6. MINIMAL PATHS IN GENERAL NETWORKS	15
REFERENCES.	19

EXPECTED CRITICAL PATH LENGTHS IN PERT NETWORKS

1. INTRODUCTION

In this note we propose a method for calculating an approximation to the expected length of a critical path in a PERT network in which arc lengths are random variables. The usual practice in this situation (as in other linear programming problems with random objective functions) is simply to replace each distribution by its expected value, thereby obtaining a deterministic problem whose solution provides an estimate g of the expected critical path length e . This estimate is always optimistic: $g \leq e$. We shall describe a way of computing a usually better, and never worse, approximation f ; this approximation satisfies $g \leq f \leq e$. The proof of this inequality is easy, once the procedure for obtaining f has been described, but the basic idea seems not to have been noted before.

The PERT model [4] is reviewed in Sec. 2. Section 3 introduces the notation used throughout the remainder of the paper and outlines sufficient conditions on the underlying distributions in order that the inequality $g \leq f \leq e$ hold. The procedure for calculating f is presented in Sec. 4, and a proof of the inequality given. Section 5 contains some numerical examples for which the numbers g, f, e are compared. The concluding Sec. 6 sketches an application to the random shortest-path problem in more general networks.

2. PERT NETWORKS

A PERT network is a directed, acyclic network. In the PERT model, such a network is viewed as representing a partial ordering of the many individual "jobs" (arcs of the network) that together comprise some "project" (the complete network), the partial ordering coming from the requirement that all inward-pointing jobs at a node must be finished before any outward-pointing job at the node can be started. Any partially ordered set can be so represented by introducing dummy arcs appropriately. We may also add to the network, if necessary, two distinguished nodes, called origin and terminal, respectively, together with appropriate outward-pointing arcs at the origin and inward-pointing arcs at the terminal, in such a way that every node is contained in some path (directed chain) from origin to terminal. The nodes of the network are thought of as "events" in time: if jobs are assigned duration times, i.e., arcs are assigned lengths, the length of a longest path from the origin to any other node of the network represents the (earliest) time of occurrence of that node. The duration time of the entire project is then the length of a longest path from origin to terminal. Such a path is called critical. We also use this terminology in speaking of a longest path from the origin to some other node.

Since the network is acyclic, it is a trivial matter to compute critical path lengths from the origin to all other nodes, given a fixed assignment of arc lengths. But if arc

lengths are random variables, it is usually not feasible to compute, precisely, the expected critical path lengths from the origin. For example, if job times are independently distributed and if each can assume one of two values, an optimistic and a pessimistic estimate, say, and if there are m jobs, then 2^m deterministic problems need to be solved. Since a typical PERT network may involve hundreds or thousands of arcs, the precise calculation of expected critical path lengths would of course be out of the question.

3. ASSUMPTIONS AND NOTATION

Suppose given a PERT network as described in Sec. 2, with an origin and terminal. Index the nodes $1, 2, \dots, n$ in such a way that 1 is the origin, n the terminal, and if there is an arc from node i to node j , then $i \leq j$. (Such a numbering can be secured as follows. The origin, with only outward-pointing arcs, is numbered 1. Delete all arcs from the origin and search for nodes in the new network that have only outward-pointing arcs; number these $2, 3, \dots, k$ in any order. In general, delete all arcs pointing out from newly numbered nodes, look for nodes with only outward-pointing arcs, and number these consecutively in any order. Eventually the terminal is numbered n . See Fig. 3.1 for an example.)

We also index the arcs $1, 2, \dots, m$ in any order and let α range over these indices.

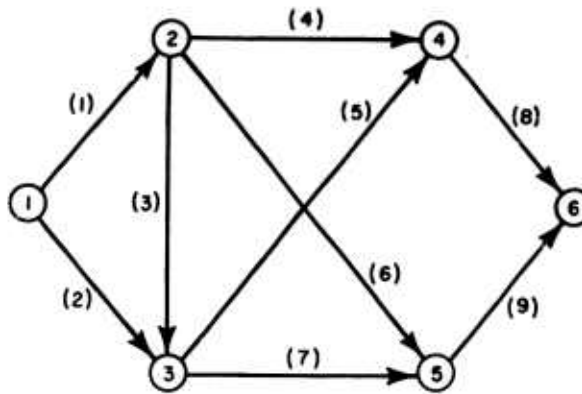


Fig. 3.1

For each node i we let B_i denote the bundle of inward-pointing arcs at node i . (Thus B_1 is always the empty set. In the example of Fig. 3.1, $B_2 = \{1\}$, $B_3 = \{2,3\}$, and so on.) We allow joint probability distributions on arc lengths within a bundle, the distributions on distinct bundles being assumed independent. Thus, corresponding to each bundle B of arcs, say

$$(3.1) \quad B = \{\alpha_1, \dots, \alpha_k\},$$

there is a nonnegative* random vector of arc-length variables

$$(3.2) \quad t_B = (t_{\alpha_1}, \dots, t_{\alpha_k}),$$

*The assumption of nonnegativity is not essential in this or the following section.

assumed to have a finite probability distribution. Taking the usual liberty with functional notation, we let

$$(3.3) \quad p(t_B)$$

represent this distribution, irrespective of the bundle.

An important special case of these assumptions is that in which individual arc lengths are distributed independently. Since we do not need this more stringent requirement, and since one can conceive of practical problems in which arc lengths within a bundle are correlated, with independence between bundles (for example, communication networks in which nodes are subject to random delay times), we have stated our assumptions as generally as possible.

For a given assignment

$$(3.4) \quad t = (t_1, t_2, \dots, t_m)$$

of lengths to all arcs of the network,

$$(3.5) \quad p(t) = p(t_{B_1})p(t_{B_2}) \cdots p(t_{B_n})$$

is the probability of this assignment, and

$$(3.6) \quad l_1(t)$$

denotes the length of a longest or critical path from node 1 to node 1, given the assignment t . Then

$$(3.7) \quad e_i = \sum_t p(t) l_i(t)$$

is the expected length of a critical path to node i .

Observe that for a given assignment t , the length $l_i(t)$ of a critical path to node i does not depend on the values assigned to arcs of the bundles B_{i+1}, \dots, B_n , since no path from 1 to i uses any of these arcs. Accordingly, if we extend the notation (3.2) to arbitrary sets S of arcs,

$$(3.8) \quad t_S = (t_{\alpha_1}, t_{\alpha_2}, \dots, t_{\alpha_s}), \quad \alpha_k \text{ in } S,$$

then (3.7) may be written as

$$(3.9) \quad e_i = \sum_{t_{A_1}} p(t_{A_1}) l_i(t_{A_1}),$$

where

$$(3.10) \quad A_1 = \bigcup_{k=1}^i B_k,$$

$$(3.11) \quad p(t_{A_1}) = p(t_{B_1})p(t_{B_2}) \cdots p(t_{B_i}).$$

Let t_α be a component of the bundle vector t_B . Then the expected length of arc α is

$$(3.12) \quad \bar{t}_\alpha = \sum_{t_B} p(t_B) t_\alpha.$$

For the arc lengths (3.12), we let g_i denote the length of

a critical path to node i . The g_i may be computed by the recursion

$$(3.13) \quad \begin{cases} g_1 = 0, \\ \dots \\ g_i = \max (g_1 + \bar{t}_1, \dots, g_{i-1} + \bar{t}_{i-1}), \quad i = 2, \dots, n. \end{cases}$$

Here we are assuming, and without loss of generality, that B_i consists of arcs from nodes $1, 2, \dots, i-1$ to i , and that, at the i -th stage in applying (3.13), these arcs are numbered $1, 2, \dots, i-1$, respectively. Of course, if the arc k (from k to i) does not exist in the network, then the corresponding term $g_k + \bar{t}_k$ of (3.13) is ignored. This is equivalent to setting such \bar{t}_k equal to $-\infty$.

4. THE NUMBERS f_i

The computation (3.13) is an easy task, even for large PERT networks having thousands of arcs. Moreover, it is not hard to see (and will be a by-product of the proof at the end of this section) that the inequalities

$$(4.1) \quad g_i \leq e_i, \quad i = 1, 2, \dots, n,$$

hold. We now describe an analogous but somewhat more complex computation, which appears to yield much closer approximations to the expected critical path lengths.

Define the numbers f_i recursively by

$$(4.2) \quad \begin{cases} f_1 = 0, \\ \dots \\ f_i = \sum_{t_{B_1}} p(t_{B_1}) \max(f_1 + t_1, \dots, f_{i-1} + t_{i-1}), i = 2, \dots, n. \end{cases}$$

As in (3.13), it is assumed in (4.2) that in computing f_i the arc numbering has been selected so that

$$(4.3) \quad t_{B_1} = (t_1, t_2, \dots, t_{i-1}),$$

with appropriate components set equal to $-\infty$.

Before proceeding further, we illustrate (3.13) and (4.2) by means of a small numerical example. Let the network be that of Fig. 4.1 below and suppose that each arc independently assumes the lengths 0 or 1, each with probability 1/2.

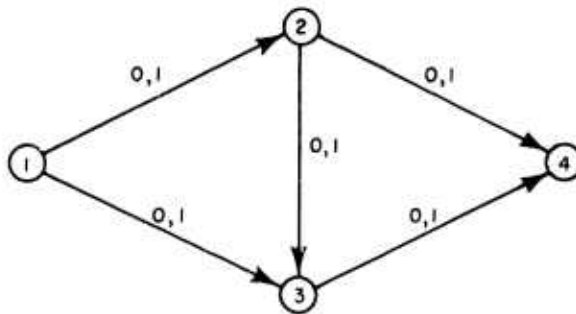


Fig. 4.1

Applying (3.13) first, we have

$$g_1 = 0,$$

$$g_2 = g_1 + \frac{1}{2} = \frac{1}{2},$$

$$g_3 = \max (g_1 + \frac{1}{2}, g_2 + \frac{1}{2}) = 1,$$

$$g_4 = \max (g_2 + \frac{1}{2}, g_3 + \frac{1}{2}) = \frac{3}{2}.$$

We next use (4.2) to obtain

$$f_1 = 0,$$

$$f_2 = \frac{1}{2}(f_1 + 0) + \frac{1}{2}(f_1 + 1) = \frac{1}{2},$$

$$\begin{aligned} f_3 &= \frac{1}{4} \max (f_1 + 0, f_2 + 0) + \frac{1}{4} \max (f_1 + 0, f_2 + 1) \\ &\quad + \frac{1}{4} \max (f_1 + 1, f_2 + 0) + \frac{1}{4} \max (f_1 + 1, f_2 + 1) \\ &= \frac{1}{4}(\frac{1}{2} + \frac{3}{2} + 1 + \frac{3}{2}) = \frac{9}{8}, \end{aligned}$$

$$\begin{aligned} f_4 &= \frac{1}{4} \max (f_2 + 0, f_3 + 0) + \frac{1}{4} \max (f_2 + 0, f_3 + 1) \\ &\quad + \frac{1}{4} \max (f_2 + 1, f_3 + 0) + \frac{1}{4} \max (f_2 + 1, f_3 + 1) \\ &= \frac{1}{4}(\frac{9}{8} + \frac{17}{8} + \frac{3}{2} + \frac{17}{8}) = \frac{55}{32}. \end{aligned}$$

An examination of the $2^5 = 32$ cases involved in calculating the expected critical path lengths e_i shows that $f_1 = e_1$ in this example.

Note that the real difference between the f -calculation and the e -calculation in the arc-independent case is that, roughly speaking, the former increases exponentially with bundle size, while the latter increases exponentially with network size. Thus the f -calculation could well be feasible in large networks.

We now show that for any PERT network, the inequalities

$$(4.4) \quad g_1 \leq f_1 \leq e_1, \quad i = 1, 2, \dots, n,$$

are satisfied. Clearly (4.4) holds for $i = 1$, and we proceed by induction. Thus we assume (4.4) for $1, 2, \dots, i$ and consider the numbers $g_{i+1}, f_{i+1}, e_{i+1}$. We show first that

$$(4.5) \quad g_{i+1} \leq f_{i+1}.$$

By (4.2), letting $B = B_{i+1} = \{1, 2, \dots, i\}$, we have

$$f_{i+1} = \sum_{t_B} p(t_B) \max (f_1 + t_1, \dots, f_i + t_i),$$

or

$$(4.6) \quad f_{i+1} = \sum_{t_B} \max [p(t_B)(f_1 + t_1), \dots, p(t_B)(f_i + t_i)].$$

Interchanging the operations of summing and taking the maximum in (4.6) yields the inequality

$$(4.7) \quad f_{i+1} \geq \max \left[\sum_{t_B} p(t_B)(f_1 + t_1), \dots, \sum_{t_B} p(t_B)(f_i + t_i) \right].$$

Hence from (3.12), noting that f_1, \dots, f_i do not depend on t_B , we have

$$(4.8) \quad f_{i+1} \geq \max [f_1 + \bar{t}_1, \dots, f_i + \bar{t}_i].$$

Consequently the induction assumption and (3.13) imply

$$(4.9) \quad f_{i+1} \geq \max [g_1 + \bar{t}_1, \dots, g_i + \bar{t}_i] = g_{i+1},$$

establishing (4.5).

The proof will be completed by showing that

$$(4.10) \quad e_{i+1} \geq f_{i+1}.$$

To simplify the notation, let $A = \bigcup_{k=1}^i B_k$, $C = \bigcup_{k=1}^{i+1} B_k$,
 $B = B_{i+1} = \{1, 2, \dots, 1\}$. Then from (3.9), (3.11), and the
 equation

$$l_{i+1}(t_C) = \max [l_1(t_A) + t_1, \dots, l_1(t_A) + t_1],$$

we have

$$e_{i+1} = \sum_{t_A} \sum_{t_B} p(t_A)p(t_B) \max [l_1(t_A) + t_1, \dots, l_1(t_A) + t_1],$$

or

$$(4.11) \quad e_{i+1} = \sum_{t_B} p(t_B) \sum_{t_A} \max [p(t_A)(l_1(t_A) + t_1), \dots, \\ p(t_A)(l_1(t_A) + t_1)].$$

It follows that

$$(4.12) \quad e_{i+1} \geq \sum_{t_B} p(t_B) \max [\sum_{t_A} p(t_A) l_1(t_A) + t_1, \dots, \\ \sum_{t_A} p(t_A) l_1(t_A) + t_1].$$

Consequently, using (3.9) again, we have

$$(4.13) \quad e_{i+1} \geq \sum_{t_B} p(t_B) \max [e_1 + t_1, \dots, e_1 + t_1].$$

The induction assumption and (4.2) now imply

$$(4.14) \quad e_{i+1} \geq \sum_{t_B} p(t_B) \max [f_1 + t_1, \dots, f_1 + t_1] = f_{i+1}.$$

This verifies (4.10) and thus proves (4.4).

It should be pointed out that the inequalities $g_1 \leq e_1$ hold under the weaker assumption that all arc lengths are distributed jointly, as is readily verified. But the assumption of bundle independence is essential for (4.4), as the following example shows. Let the network be that of Fig. 4.2 below, with the distribution of arc lengths given in the accompanying table.

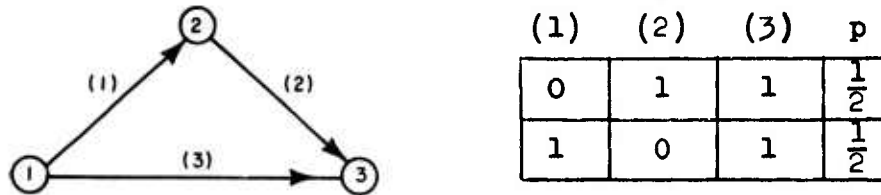


Fig. 4.2

Then $e_3 = g_3 = 1$, but $f_3 = 5/4$.

We also remark that all of the various logical possibilities implicit in (4.4) can occur. It is not difficult to construct examples in which

$$(4.15) \quad g_n < f_n < e_n,$$

$$(4.16) \quad g_n < f_n = e_n,$$

$$(4.17) \quad g_n = f_n < e_n.$$

$$(4.18) \quad g_n = f_n = e_n.$$

For instance, the network of Fig. 4.2 with the bundle-independent distribution

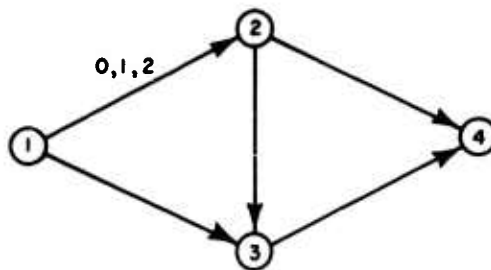
(1)	(2)	(3)	p
0	0	0	$\frac{1}{4}$
0	0	1	$\frac{1}{4}$
2	0	0	$\frac{1}{4}$
2	0	1	$\frac{1}{4}$

satisfies (4.17): $g_3 = f_3 = 1 < e_3 = 5/4$.

5. SOME NUMERICAL EXAMPLES

In this section we present some numerical examples for purposes of comparing the numbers g_n, f_n, e_n . In each of the examples, we have assumed the same uniform distribution on each arc, distributions on distinct arcs being independent. The numbers recorded beside a typical arc are the lengths that can be taken on. In Examples 2 and 3, the computation of e_n was carried out on a computer, using a program written by R. Clasen.

Example 1.

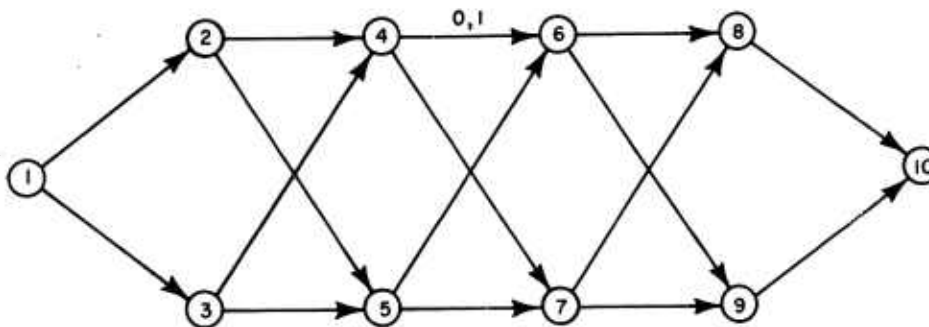


$g_4 = 3.00, f_4 = 3.22, e_4 = 3.32.$

Distribution of critical path length:

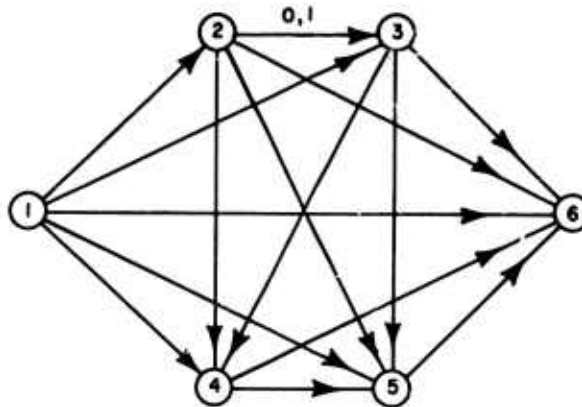
l_4	0	1	2	3	4	5	6
freq.	1	11	50	74	71	27	9

Example 2.



$g_{10} = 2.50, f_{10} = 3.25, e_{10} = 3.95.$

Example 3.



$$g_6 = 2.50, f_6 = 2.92, e_6 = 2.95.$$

6. MINIMAL PATHS IN GENERAL NETWORKS

If one is interested in expected minimal path lengths in acyclic directed networks,* the theory of Sec. 4 remains valid by replacing "max" by "min" and reversing all inequalities. Suppose we depart from the PERT context completely and consider general directed networks whose arc lengths are nonnegative random variables. What can be said about expected maximal or minimal path lengths from some node to all others in this case?

Since the problem of determining a maximal-length simple path between two given nodes in the deterministic case is no

*The warehousing problem with random costs can be formulated in this way [3]. Also, a number of stochastic dynamic programming problems have such formulations.

longer easy, being equivalent to a traveling-salesman problem, we restrict attention to the minimal-path case, for which the character of the deterministic problem is still much the same [1,2].

Although it can be shown that the analogues of equations (4.2) have a solution, and in fact, under fairly mild conditions on the network and the distributions involved, a unique solution, it is no longer true that the analogous inequalities $e_1 \leq f_1$ hold for this solution. The following example demonstrates this:

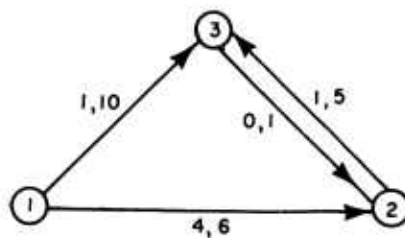


Fig. 6.1

In Fig. 6.1 we are assuming that the arc lengths written adjacent to an arc are equiprobable, with distributions on distinct arcs independent. The unique solution of the equations

$$f_1 = 0,$$

$$f_2 = \frac{1}{4} \min (f_1 + 4, f_3 + 0) + \frac{1}{4} \min (f_1 + 4, f_3 + 1) \\ + \frac{1}{4} \min (f_1 + 6, f_3 + 0) + \frac{1}{4} \min (f_1 + 6, f_3 + 1),$$

$$f_3 = \frac{1}{4} \min (f_1 + 1, f_2 + 1) + \frac{1}{4} \min (f_1 + 1, f_2 + 5) \\ + \frac{1}{4} \min (f_1 + 10, f_2 + 1) + \frac{1}{4} \min (f_1 + 10, f_2 + 5),$$

is

$$f_1 = 0, \quad f_2 = 4^{1/3}, \quad f_3 = 4^{1/6}.$$

But the expected minimal path lengths from node 1 are

$$e_1 = 0, \quad e_2 = 3^{1/4}, \quad e_3 = 4^{3/8}$$

for this example.

The inequality fails to hold in networks with cycles because the distribution of minimal path lengths to beginning nodes of the bundle of arcs pointing into a given node can depend on the distribution of arc lengths for the bundle. However, approximations f'_i that still satisfy

$$(6.1) \quad e_i \leq f'_i \leq g_i$$

can be calculated by the following device. First assign each arc its expected length and compute the numbers g_i representing the minimal path length from 1 to i under this assignment. Next renumber the nodes according to increasing

g_1 , keeping the origin numbered 1. Now delete all arcs that lead from higher-numbered nodes to lower-numbered nodes, restricting the probability distributions to the remaining arcs. The sub-network left is acyclic and the numbers e_1^i , f_1^i , g_1^i for it satisfy

$$(6.2) \quad e_1^i \leq f_1^i \leq g_1^i.$$

Now $e_1 \leq e_1^i$, since the e_1^i come from a sub-network of the original network. But note that equality holds in the corresponding inequality $g_1 \leq g_1^i$. Thus we have

$$(6.3) \quad e_1 \leq e_1^i \leq f_1^i \leq g_1^i = g_1^i,$$

which yields (6.1).

REFERENCES

1. Bellman, R., "On a Routing Problem," Q. Appl. Math., Vol. 16, 1958, pp. 87-90.
2. Dantzig, G. B., "Discrete Variable Extremum Problems," Op. Res., Vol. 5, 1957, pp. 266-277.
3. ———, "On the Status of Multi-stage Linear Programming Problems," I.S.I. Bulletin, Vol. 36, pp. 303-320.
4. Malcolm, D. G., J. H. Roseboom, C. E. Clark, and W. Fazar, "Application of a Technique for Research and Development Program Evaluation," Op. Res., Vol. 7, 1959, pp. 646-669.