NASr-21(08)

MEMORANDUM
RM-4581-NASA
SEPTEMBER 1965

A NETWORK APPROACH TO PARTS PROVISIONING FOR APOLLO PRELAUNCH OPERATIONS

R. B. Brooks and R. H. Haase

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

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

Published by The RAND Corporation

PREFACE

This Memorandum is addressed to the problem of establishing a replacement-parts policy for Apollo prelaunch operations. It develops a network technique to help determine economic stock levels and resupply capabilities that will minimize the deleterious effects of parts shortages on the schedule of operations. Though the study is oriented toward Apollo prelaunch operations, the techniques also should be of interest to those personnel concerned with establishing replacement-parts policies for projects other than Apollo.

Sections I, II, and III present the basic network approach, which is later illustrated in Sec. IV with numerical examples. The computational techniques developed for computer implementation of the approach are described in the Appendix.

The Apollo Checkout System Study is a continuing program to help define checkout system and support plans for the Apollo mission. As one of a series documenting the Study, this Memorandum was prepared by The RAND Corporation for Hq, National Aeronautics and Space Administration under Contract NASr-21(08).

BLANK PAGE

SUMMARY

The Apollo prelaunch operations consist of a sequence of activities, some of which result in the discovery of malfunctions that may, in turn, result in demands for replacement parts for the vehicle. These parts may be available from stock on hand, or from such sources as local bench repair of malfunctioning parts, local assembly from components, subsequent vehicles, and the manufacturer. The purpose of this Memorandum is to develop methods that will assist in establishing a strategy -- a replacement-parts policy -- governing the use of these sources.

To select from among replacement-parts policies or even to evaluate one policy, it is necessary to have measures of both cost and effectiveness. The measure of cost must reflect at least the cost of establishing and maintaining the stock levels and resupply capability. The measure of effectiveness should be formulated in terms of the amount of delay to the schedule.

Selecting and evaluating a replacement-parts policy for projects such as Apollo prelaunch operations has two special features that differentiate it from other inventory problems. First, a project consists of an operations plan that specifies the sequence of activities or tests to be performed. It is during these tests that malfunctions are identified and demands for parts are generated. Once the operations plan is specified, therefore, it should be possible to identify the locations in the schedule where demands for a particular part might occur. Second, the relationship between parts shortages and effectiveness is complicated. It is easy to construct examples of parts shortages that cause little or no delay in the schedule because subsequent activities can continue without the parts. Contrariwise, it is possible to construct examples of shortages that stop all operations. Actually, the effect of a parts shortage on delay depends not only on how long the shortage lasts, but also on where in the schedule the demand occurs and where in the schedule the demand must be filled. The effect one shortage has may also be modified if other shortages occur.

The approach used in this Memorandum represents the scheduled operations as a project network. Unscheduled activities are also

performed, however, as the project unfolds. For present purposes, the unscheduled activities are those associated with replacing malfunctioning parts, e.g., bench repair. These activities are represented by adding arcs to the nominal network. The way these arcs are added and the times associated with them are functions of the particular parts policy being evaluated.

Data required by this approach are a description of the schedule of operations in project network form, a list of parts, and a list of "possible demands." Each part is characterized by the quantity stocked and by the time required to repair a defective part (or, more generally, the resupply time). Each possible demand is characterized by the point in the network at which it can occur and the point by which it must have been met, by identification of the part demanded, and by the probability of the demand.

The following procedure is used for each resupply (e.g., bench repair) capability. First, reduce the size of the problem by identifying parts that do not need to be stocked and aggregating some of the activities in the nominal schedule. Next, use marginal analysis to determine a sequence of stock levels for the given resupply capability. Then estimate the expected measure of effectiveness for each replacement-parts policy in the sequence and develop the corresponding cost/effectiveness curve. Repeat this procedure for each resupply capability of interest, and then select a replacement-parts policy by choosing a point on one of the curves. The procedure is illustrated in Sec. IV, using computer programs (including a Stock Selection and an Evaluation Model) whose algorithms are described in the Appendix.

ACKNOWLEDGMENTS

We are indebted to a number of our colleagues at RAND for their assistance and encouragement in the preparation of this Memorandum. We would like especially to thank A. J. Clark, S. I. Firstman, and R. J. Freeman. We are grateful also to Mr. J. L. Marchbanks of the San Antonio Air Materiel Area for supplying the data on which the examples in Section IV are based.

BLANK PAGE

APOLLO CHECKOUT SYSTEM STUDY BIBLIOGRAPHY

The Apollo Checkout System Study is a continuing research program conducted by The RAND Corporation, under Contract NASr-21(08), to assist the National Aeronautics and Space Administration in defining an integrated checkout system and in developing associated mission and support plans for the Apollo program. Results of the study have been reported in this and the following Memoranda.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

RM-4513-NASA Scheuer, E. M., Two Models for Time Constrained Maintenance, April 1965.

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

CONTENTS

I. INTRODUCTION

The Apollo* prelaunch operations consist of a sequence of activities. Typically, these activities are inspect, transfer , assemble, disassemble, test, weigh, and checkout. Some of these activities result in the discovery of malfunctions. These malfunctions may result in demands for replacement parts for the vehicle. Many of these parts are high-cost items with limited resupply capability and relatively long reorder time. Furthermore, the facilities required to repair these parts usually are expensive. In general, replacement parts are available from such sources as stock on hand, local bench repair of malfunctioning parts, local assembly from components, subsequent vehicles, and remote sources such as the manufacturer. The purpose of this Memorandum is to present methods that will assist in establishing a strategy governing the use of these sources. Such a strategy is referred to as a replacement-parts policy. More specifically, in projects such as the Apollo prelaunch operations, the proposed methods should be useful for such problems as setting initial stock levels for parts, establishing the repair facilities that should be provided, estimating the extent to which reorder capability from the manufacturer can be substituted for local stocks, evaluating the potential effectiveness of methods for expediting orders, analyzing the effect of different stock locations, and dealing with similar problems that originate in the task of selecting and evaluating a replacement-parts policy.

During the prelaunch operations, demands for parts for the launch vehicle and spacecraft will tend to decrease as the schedule approaches the launch date, since as the schedule progresses the "bugs" should be worked out of the system. On the other hand, the penalty associated with not having a part when it is needed tends to increase as the launch date approaches, since there is less opportunity to schedule around delays. For example, a one-day delay at final countdown could scrub

---

*This includes the Apollo Spacecraft and the Saturn V Launch Vehicle.

the scheduled launch, but a similar delay while the Saturn V is in the Vertical Assembly Building or while the spacecraft is in the Operations and Checkout Building may require only minor changes in the schedule. Even though the penalty for not completing the countdown within the launch window constraints is severe, a parts policy should be formulated on the basis of considering the total operations plan. That is, it is not sufficient to look at only the countdown phase, or at any other <u>single</u> phase of the operation.

While looking only at countdown might give information about stock levels for spares, it would not be very illuminating on questions of bench repair facilities, priorities, reorder schedules, or rules for cannibalization. Bench repair facilities, for instance, are relatively useless during final countdown, because there is simply not sufficient time to bench repair a malfunctioning part. Even in setting stock levels, it is necessary to consider all phases. For example, during countdown there is limited access to the space vehicle. This means that there are relatively few parts which can be removed and replaced. After cryogenic loading, there is practically no access for maintenance activities. There is limited opportunity to remove and replace a few parts. The decision to stock these parts, which generally are small and relatively inexpensive, probably can be made without any additional analysis. However, countdown considerations alone would set only lower bounds on stock levels for these parts, since they would not take into account the more frequent demands which occur during preceding phases.

In order to select from among replacement-parts policies or even to evaluate one policy, it is necessary to have both a measure of cost and a measure of effectiveness. The measure of cost must reflect at least the cost of establishing and maintaining the stock levels and repair facilities. The measure of effectiveness should be formulated in terms of the amount of delay to the schedule (e.g., expected project time, probability of launching on schedule, or measures that depend upon the completion time of several activities, such as the mating of the spacecraft to the launch vehicle as well as countdown). A more complete discussion of measures of cost and effectiveness is given in Sec. III.

The problem of selecting and evaluating a replacement-parts policy for projects such as Apollo prelaunch operations has two special features that differentiate it from other inventory problems. These are:

1) Partial predictability of demands. An operation plan specifies the sequence of activities or tests to be performed. It is during these tests that malfunctioning parts are identified and demands for these parts are generated. Therefore, once the operations plan is specified, it should be possible to identify the locations in the schedule where demands for a particular part might occur. For example, a demand for a LOX valve may occur during functional checkout of a propulsion system, but almost certainly not during checkout of a guidance system.

2) Complicated relationship between parts shortages and effectiveness. The effect of a parts shortage on delay depends not only on the length of time the shortage exists, but also on where in the schedule the demand occurs and where in the schedule the demand must be filled.

Suppose, for example, that a portion of a project consists of checking a two-component subsystem, where test R checks for the acceptability of component r and test S checks for component s. Suppose, further, that tests R and S cannot be performed simultaneoulsy and that the operations plan specifies test sequence R, then S. Following the completion of tests R and S, the subsystem is transferred to another location. Each test takes 2 days to complete, and the transfer takes 1 day. If a test discovers a malfunctioning component, the component is sent to bench repair, and it must be replaced before the transfer activity can begin. For simplicity, suppose that after the component is repaired it is ready to be installed, and it can be installed without interfering with the other test. Suppose also that the complete test does not have to be repeated. (Actually, allowing for retest does not change the results -- only the complexity of the analysis.)

Figure 1 represents a bar chart of the present schedule of activities if both tests are "Go." The total project time is 5 days. Figure 2 represents the schedule if test R is "No Go," test S is "Go," and component r is not in stock, so that the defective component must be repaired. The project takes 5 days to complete because the 1-day repair of component r can be accomplished while test S is being conducted. Figure 3 shows the schedule if test R is "Go," test S is "No Go," and component s is not in stock. The total project time is 6 days.

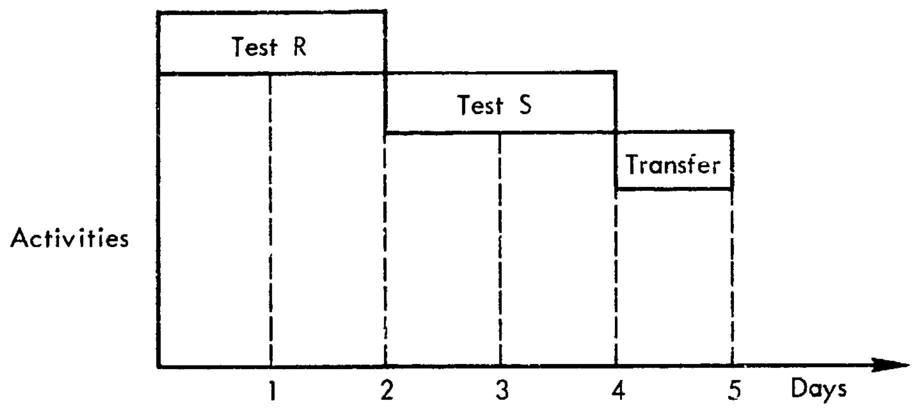


Fig. 1—Bar chart: both tests "GO"

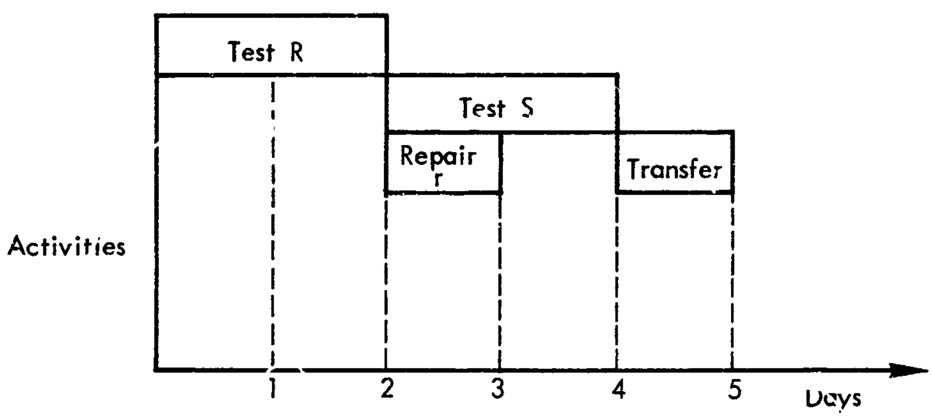


Fig. 2—Bar chart: R "NO GO", S "GO"

For the two latter situations, a parts shortage existed for 1 day; however, the shortage of component r did <u>not</u> cause a delay, while the shortage of component s <u>did</u> cause a 1-day delay in the transfer activity. Actually, the effect of a parts shortage on delay is more complicated than this example indicates because the delay depends not only on the length of time the shortage exists, and where in the schedule the demand must be filled, but also upon the occurrence of other shortages for the same part as well as for other parts.

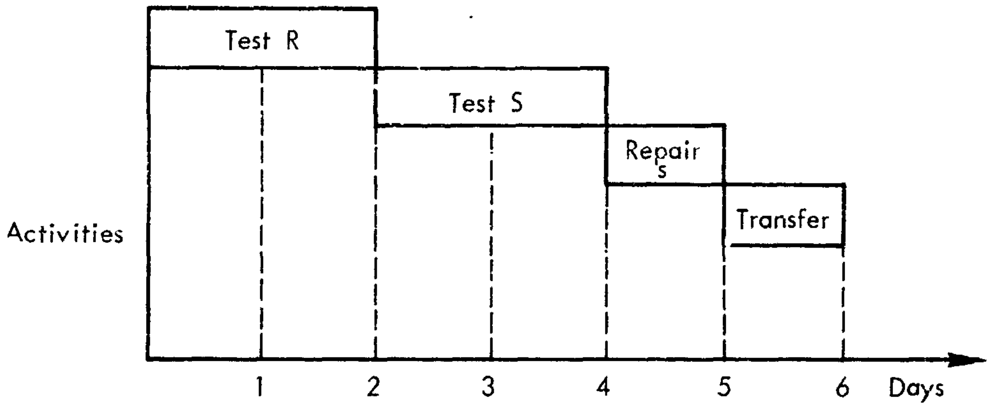


Fig.3—Bar chart: R "GO", S "NO GO"

Any method used to evaluate and select replacement-parts policies for Apollo prelaunch operations at Merritt Island should be consistent with the two special features described above. That is, the methods should account for the complicated relationship between parts shortages and delay, and should make use of the partial predictability of demands. In so doing, they will not only make use of such traditional data as malfunction rates, costs of stocking parts, and repair or reorder times, but also data on the schedule of operations.

In Sec. II, where the basic network approach is described by means of a simple example, the primary concern is with presenting the concepts embodied in the network approach, rather than the specific assumptions required. Section III states these assumptions, shows how some of them can be relaxed, and discusses some of the limitations of the approach. Two numerical examples are presented in Sec. IV. Computational techniques, which may be used to implement the approach and which form the basis for the computer program, are described in the Appendix.

II. DESCRIPTION OF A NETWORK APPROACH

As Sec. I indicates, the problem of establishing a replacement-parts policy for Apollo prelaunch operations is distinguished from other inventory problems by certain unique characteristics. The purpose of Sec. II is to describe a network approach to this problem which incorporates the two special features identified in Sec I: 1) the partial predictability of demands, and 2) the complicated relationship between parts shortages and effectiveness.

The chief concern in Sec. II is with the description of the approach itself, rather than of the ways in which it can be applied. Emphasis upon the approach and upon the concepts is here stressed through the use of some simplifying assumptions and the limiting of discussions to elementary examples. The statement and discussion of underlying assumptions is postponed until Sec. III. After an examination of the problem of evaluating a replacement-parts policy, the problem of selecting a policy is considered. The final portion of this section discusses two ways of reducing the size of the problem.

A METHOD FOR EVALUATING A REPLACEMENT-PARTS POLICY

The approach to evaluating a replacement-parts policy presented here is to represent the scheduled prelaunch operations as a project network.* During these scheduled assembly and checkout operations, various non-scheduled activities are performed. For present purposes, the important non-scheduled activities are those associated with replacing malfunctioning parts. These replacement activities are included by means of adding arcs to the original network. The manner in which these arcs are added and the times associated with them are functions

---

*For a discussion of project networks, see, for example, J. E. Kelley and M. R. Walker, "Critical Path Planning and Scheduling," Proceedings of the E.J.C.C., 1959, pp. 160-172; and D. G. Malcolm, J. H. Roseboom, C. E. Clark, and W. Fazar, "Application of a Technique for Research and Development Program Evaluation," ORSA, Vol. 7, No. 5, September-October, 1957, p. 646.

of the particular parts policy being evaluated. An example will demonstrate the procedure for evaluating a parts policy.

Suppose that Fig. 4 represents a schedule of activities in project network form for a two-part system. There are a total of eight activities represented by arcs in the network. Each of these activities is a scheduled operation, such as a checkout, an assembly, or a transfer. The nodes, which are indicated by the encircled numerals, represent events or specific points in time.

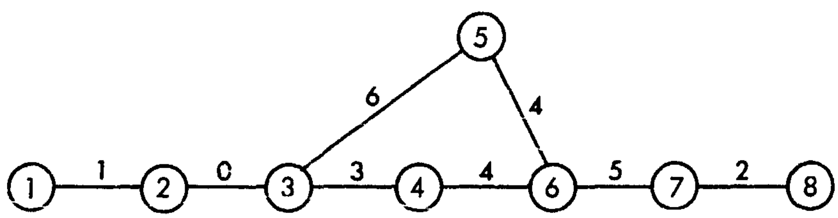


Fig. 4—Nominal schedule

The network shows precedence relations among activities. For instance, activity (6-7) cannot be started until both activities (4-6) and (5-6) are completed. But activity (3-5) can be performed simultaneously with activity (3-4). In addition, each activity has a time associated with it which represents the time required to perform the activity. These times are shown above the arcs in Fig. 4. The data needed to construct Fig. 4 are given in Table 1. Note that activity (2-3) requires zero days to perform. This is a dummy activity that is required for demand interpretation; it is explained below.

Table 2 describes all of the possible demands which can occur for these two parts. The "demand node" identifies the point in the network at which the demand occurs, if, in fact, it does occur. The "fill node" identifies the point in the network at which the demand must be filled in order to avoid delaying subsequent activities. The dummy activity

(2-3) serves to represent the situation that part R must be replaced immediately, because effectively the demand node and the fill node occur simultaneously. It is assumed that demands are independent events with the indicated probabilities.

Table 1

ACTIVITIES LIST

First Node	Second Node	Time Required
1	2	1
2	3	0
3	4	3
3	5	6
4	6	4
5	6	4
6	7	5
7	8	2

Table 2

POSSIBLE DEMANDS

Identification Number	Part Type	Demand Node	Fill Node	Probability of Demand
1	R	2	3	.1
2	R	4	6	.2
3	S	5	7	.3

The replacement-parts policy to be evaluated here consists of providing one spare for part R and no spares for part S, and bench repair facilities capable of repairing part R in 11 days and part S in 13 days. This parts policy is represented in Table 3.

The parts policy for this example has been evaluated in Table 4. Since there are three possible demands, there are $2^3 = 8$ different sets of demands which can occur. Each of these sets is listed in the first column of Table 4. The second and third columns of the table contain, respectively, the project time associated with each of these

Table 3

PARTS POLICY

Type	Quantity in Stock	Repair Time
R	1	11
S	0	13

Table 4

EVALUATION OF PARTS POLICY

Realized Demands	Project Time	Probability	Probability x Project Time
None	18	.504	9.072
1	18	.056	1.008
2	18	.126	2.268
1,2	19	.014	0.266
3	22	.216	4.752
1, 3	22	.024	0.528
2,3	22	.054	1.188
1,2,3	22	.006	0.132
			19.214[a]

[a]Expected time in days.

possible demand patterns and the probability of occurrence. To demonstrate the evaluation procedure, consider, respectively, the first, the fifth, the fourth, and the last rows of the table.

1) Demand Set = {none}. When there are no demands, the only activities are the scheduled ones--that is, there are no non-scheduled activities. Thus, the network which must be evaluated is that given in Fig. 4. Evaluating a project network consists of attaching a time to each node in the network; this time represents the earliest time that the activities having that node for their first node can be begun. The time assigned to the last node of the network is the project time. The procedure is to assign the time 0 to the first node of the network,

and to proceed as follows to assign times to the remaining nodes. For the network shown in Fig. 4, the time for the second node is the time for the first node plus the activity time of one day for a total of (0+1) = 1 day. Similarly, the time at node 3 is (1+0) = 1, at node 4 is (1+3) = 4, at node 5 is (1+6) = 7. The time at node 6 is the greater of (4+4) or (4+7), i.e., 11 days. The time at node 7 is (11+5) = 16, and, finally, the total project time is (16+2) = 18 days. Note that if all activities are on schedule, activity (4-6) is completed three days ahead of activity (5-6). Thus, there is a three-day slack period for activity (4-6), which means that a three-day delay in the completion of this activity does not delay activity (6-7) or the project. The total project time of 18 days is for the case where there are no demands. If the independence assumption and the probabilities given in Table 2 are used, the probability of no demands is the probability that demands 1, 2, and 3 do not occur, i.e., (1-.1)x(1-.2)x(1-.3) =.504.

2) Demand Set = {3}. Now consider the case where the only demand that occurs is demand number 3(i.e., a demand for part S occurring at node 5 that must be filled by node 7). This demand gives rise to one non-scheduled activity, namely, the removal, bench repair, and replacement of the malfunctioning part. This activity cannot begin until node 5 and must be completed by node 7. Thus, the network representation of the operations that must be performed, both scheduled and non-scheduled, is as shown in Fig. 5. An arc from node 5 to node 7 is

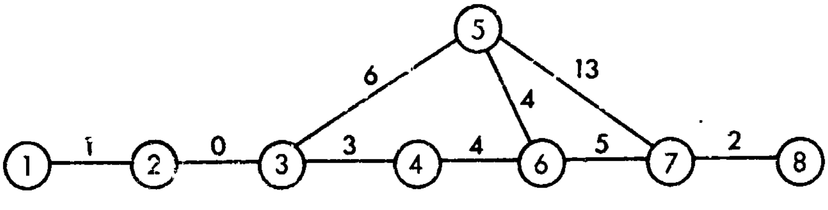


Fig 5—Augmented network (one demand)

added to the nominal network given in Fig. 4. Assuming that the removal and replacement of parts takes negligible time, the time associated with this added activity is the bench repair time for part S, namely, 13 days. (See Table 3.) The evaluation of this network gives a total project time of 22 days. Note that the increase over the nominal project time (Fig. 4) is 4 days and is not the same as the bench repair time for the part, which is 13 days. This demonstrates the necessity of evaluating a parts policy in the context of the schedule of operations. Again, under the assumption of independence of demands and for the probabilities given in Table 2, the probability of only demand 3 occurring is (1-.1)x(1-.2)x(.3) = .216.

3) <u>Demand Set = {1,2}</u>. Figure 6 shows the augmented network when demand 1 and demand 2 occur (i.e., a demand for part R at node 2, which must be filled by node 3, and another demand for part R, which occurs at node 4 and must be filled by node 6). The demand that occurs at node 2 can be filled from stock, since one of part R is stocked. This demand gives rise to a remove-and-replace activity the first node of which is 2, and the second 3. The activity will take 0 time, since, for this example, it is assumed that removal and replacement time is negligible. Although there is not another part in stock to fill the

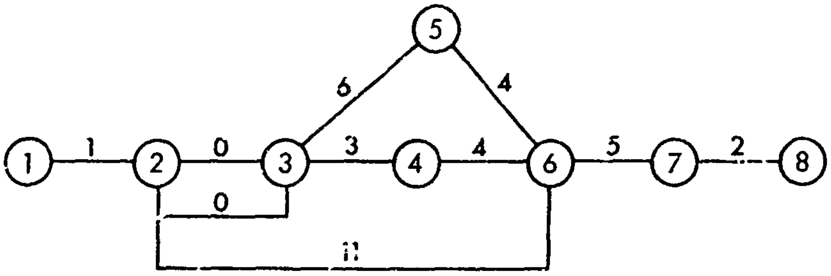


Fig. 6—Augmented network (two demands)

second demand, the part that is removed at node 2 can be repaired and used to fill the demand occurring at node 4. Thus, the second demand generates a remove, bench repair, and replace activity the first node of which is 2 and the second node 6. The activity has a time of 11 days--i.e., the repair time for part R.

The total project time for the network is 19 days, which represents an increase of 1 day over the nominal project time. Again, this could not have been inferred from the repair times alone. The probability that demands 1 and 2 but not 3 occur is (.1)x(.2)x(1-.3) = .014.

4) Demand Set = {1,2,3}. When all demands occur, three nonscheduled activities must be added to the nominal network. They are the same activities identified under 2) and 3) above. The augmented network is shown in Fig. 7. The total project time is 22 days. Note that this project time is the same as when only demand 3 occurs even though demands 1 and 2 alone cause a one-day increase in the nominal schedule.

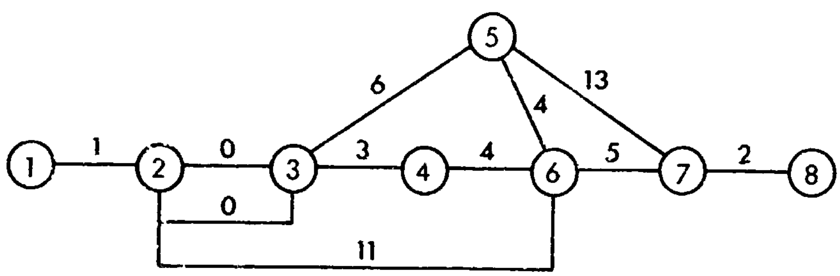


Fig. 7—Augmented network (all demands)

The probability that all three demands occur is (.1)x(.2)x(.3) = .006.

The project times listed in Table 4 with their associated probabilities give an expected project time for the parts policy of 19.214 days. Project time is only one of many possible measures of effectiveness that can be used. For instance, one may be interested in the probability of completing the project within a specified number of days or of hitting the launch window. Also, for subsequent scheduling, the transfer of the space vehicle from the Vertical Assembly Building to the launch pad may be an important event but still less critical than the actual launch. In this case a combined measure can be used that is some weighted combination of the time of the transfer and the time of the launch. In the case of a multi-vehicle project the measure may be a function of the launch times for several vehicles. A more complete discussion of measures of effectiveness is presented in Sec. III.

The computer program (Evaluation Model) which was written for this Memorandum takes as inputs a schedule of activities (Table 1), a parts list (Table 3), and a list of possible demands (Table 2). The output of the program is an estimate of expected effectiveness. The program evaluates a given parts policy in much the same way as that just described for the example network. However, it does not enumerate and evaluate all possible demand sets. This would be an impracticable task for any but the most trivial problem. For instance, 30 parts, each with 10 possible demands, represent 2^{300} different demand sets. Therefore, instead of enumerating all demand patterns, the procedure is to employ a Monte Carlo technique. Specifically, a sample of demand sets is drawn, and the expected measure of effectiveness is estimated from the sample. One iteration of the model proceeds as follows. First a set of demands is generated on the basis of the possible demands and their probabilities. As a function of these demands and the parts that are carried in stock, arcs that are needed to fill the demands are added to the nominal network. Times for the nodes of the network are computed. The measure of effectiveness is computed as a function of these times. (For the above example, this is the time at the last node.) The measure is then averaged with the measures obtained on previous iterations. This sequence is repeated for as many iterations as desired. A more complete description of the computations within the Evaluation Model is given in the Appendix.

SELECTING A REPLACEMENT-PARTS POLICY

The network model described above evaluates a given replacement-parts policy. Ideally, one would like to have a procedure that would not only evaluate a parts policy but would also determine an "optimum" or at least a "good" parts policy. Since a replacement-parts policy specifies both the repair facilities (or, in general, the resupply capability) and the stock levels, and since the "optimum" stock levels depend on the available repair facilities, the procedure is to select stock levels for a given repair facility. Different replacement-parts policies can be established by considering other repair facilities and by selecting stock levels that are compatible with these repair facilities.

The procedure described here to determine stock levels is an application of marginal analysis. The procedure is as follows. Let the first policy in the sequence consist of carrying no parts in stock. Then find a part which, when added to stock, results in the greatest additional expected effectiveness per added cost. This stock represents the second policy in the sequence. Subsequent policies are obtained by repeating the process. (A more complete discussion is presented in the Appendix).

Suppose one wants to generate a sequence of "good" stocking policies for the illustrative two-part project described in the preceding section. For convenience, project time will be used as the measure of effectiveness. The first step is to determine the expected project time when no parts are carried in stock, that is, when one relies on only the repair capability. Next one needs to determine the additional expected effectiveness (i.e., expected reduction in project time) per cost when adding respectively one each of parts R and S to stock. Then one adds to stock that part which gives the greatest reduction in expected project time per cost. Using the schedule of activities (Table 1) and the resulting nominal schedule (Fig. 4), the parts list (Table 3) and the list of possible demands (Table 2), one evaluates expected project time for each of the three stocking policies (i.e., respectively, none in stock, only one of part R in stock, and only one of part S in stock) in the same manner as indicated in Table 4. The results are shown in Table 5. For this example, the Stock Selection

Table 5

EXPECTED PROJECT TIME

Stocking Policy	Expected Time in Days
None	20.360
R	19.214
S	19.900

Index is the change in expected project time divided by the cost of the part. Suppose that part R costs $10 and that part S costs $12. Then the "None" column in Table 6 gives the stock selection indexes for parts R and S since (see Table 5) stocking one of R reduces the expected project time by 1.646 days and stocking one of S reduces the expected project time by .960 days. Dividing these respective changes

Table 6

STOCK SELECTION INDEXES

Part	Stocking Policy			
	None	R	R&S	R&S&R
R	.1646	.0014	.0020	0
S	.0800	.0995	0	0

by the cost of the part yields the values shown in the "None" column, which represent the change in expected project time per dollar invested. Part R is stocked on the basis of this selection index; this represents the second stocking policy in the sequence, the first being "None." The sequence is continued by repeating the above process, except that now the change in the expected project time is evaluated when one of each part is added to an existing stock of one part R. Therefore, the value .0014 in the second column represents the change in effectiveness per dollar by adding a second part R to stock and, similarly, .0995 represents the effect of adding the first part S to stock given one of R in stock. By sequentially adding to stock that part which has the largest selection index a sequence of "good" stocking policies is generated. For this example the sequence is None, R, R&S, and R&S&R.

This procedure generates, for a given bench repair facility, a sequence of stocking policies with corresponding costs and expected project times as shown in Table 7. The summary data are plotted in Fig. 8 in terms of expected delay which is defined as the difference between expected project time and the nominal project time of 18 days.

Table 7

SUMMARY DATA OF STOCKING POLICIES

Stocking Policy	Expected Project Time (Days)	Stocking Cost (Dollars)
None	20.860	00
R	19.214	10
R&S	18.020	22
R&S&R	18.000	32

If the cost of the repair facilities is included, a cost/effectiveness curve for the resulting replacement-parts policy (i.e., repair facilities and stocking policies) can be plotted. In general, different investments in repair facilities can be made, the result being different repair capabilities and repair times. Different repair facilities will result in different stock levels and, therefore, in

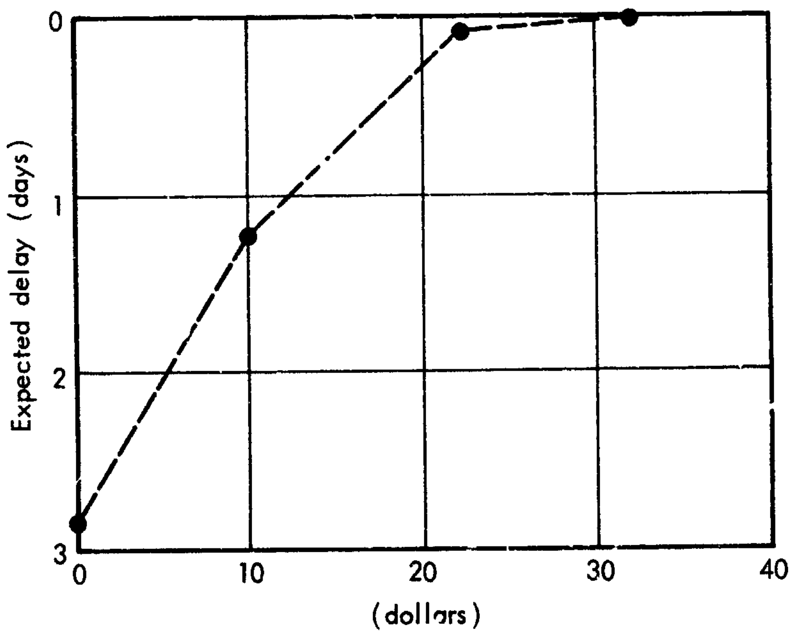


Fig.8—Cost/expected delay curve of stocking policies

different cost/effectiveness curves. Comparing these curves, and taking into account the cost of the repair facilities as well as stocks, a simultaneous choice of bench repair facilities and stocking policies (i.e., a replacement-parts policy) can be made. Suppose the cost/effectiveness curves in Fig. 9 are the result of considering three different repair facilities. Moving from curve 1 to 3 represents increasing investment in repair facilities. For a given curve, increasing investment corresponds to increasing expenditures for stocks. If, for example, one decides to invest I dollars on a replacement-parts policy, one would select bench repair policy 1 and the stocking policy represented by point a on curve 1. However, if one decides to invest I' dollars, one would select bench repair policy 2 and the stocking policy represented by point b.

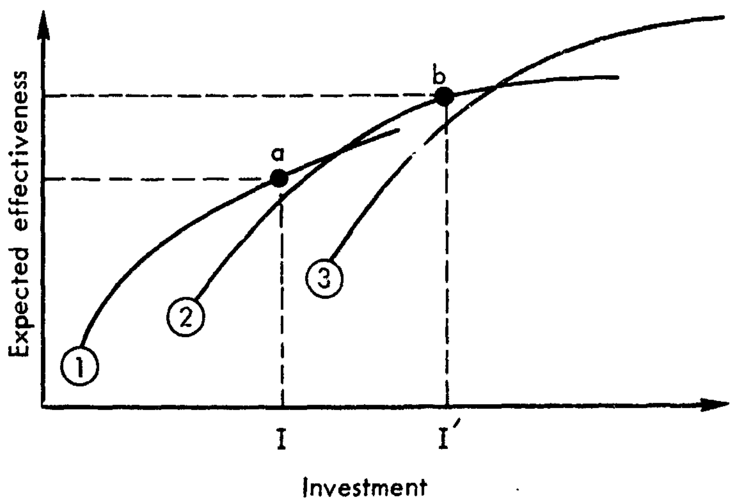


Fig. 9—Generalized cost/effectiveness curves for replacement-parts policies

A Monte Carlo computer program (the Stock Selection Model) has been written to estimate stock selection indexes. Inputs are the same as those required for the Evaluation Model, with the addition of costs for the parts and a specification of the parts whose selection indexes are to be estimated. In broad outline, one iteration of the program runs as follows. First, a set of demands is generated on the basis of the possible demands and their probabilities; these are inputs to the program. On the basis of these demands and the parts that are carried in stock, arcs that represent activities needed to fill the demands are added to the nominal network. Times for the nodes of the augmented network are then computed, and a measure of effectiveness u_0 is calculated. Then the quantity of the part being analyzed is increased by one, the demands are reinterpreted, the node times for the new network are computed, and a measure of effectiveness u_1 is computed. The difference $u_1 - u_0$ represents the incremental effectiveness of increasing by one the quantity stocked of the part being analyzed for the particular demands that were generated. This increment is then averaged in with the increments obtained on previous iterations. This process is repeated for as many iterations as desired. The estimate of the expected incremental effectiveness thus obtained is divided by the incremental cost of adding one of the part to stock--to obtain the stock selection index for that part. A more complete description of the computations within the Stock Selection Model is given in the Appendix.

REDUCING THE SIZE OF THE PROBLEM

In an attempt to keep computer running times within tolerable limits, the following techniques have been incorporated to reduce the size of the problem; 1) eliminating non-critical parts and demands, and 2) reducing the size of the network.

Eliminating Non-Critical Parts and Demands

It is quite possible that, for a given repair capability and a schedule of activities, it is not necessary to stock a particular part. That is, the repair facilities can repair a malfunctioning part within the allowable time to prevent delaying subsequent activities. Such parts are referred to a non-critical* parts, and the stock level for these parts is set at zero without further analysis.

Suppose Fig. 10 represents a nominal network. The activity times are indicated above each arc. The three dashed lines, r, s, and t, represent possible demands for three parts, R, S, and T, which have respectively repair times of 6, 8, and 8 days (for a specified bench repair capability). Comparison of the repair time with the longest

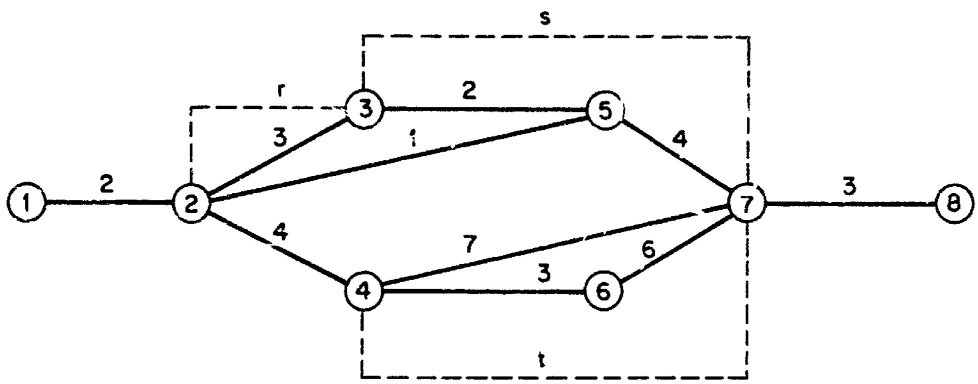


Fig. 10—Nominal network

path in the nominal network between the demand node and fill node for demand t identifies part T as a non-critical part. For t the demand node is 4 and the fill node is 7. The longest path from 4 to 7 is 9 days; therefore, the 8-day repair time for part T is adequate to prevent delaying activity (7-8). Thus, there is no need to stock part T and no need to include demand t in subsequent analysis. This is not

---

*The distinction between critical and non-critical parts is made only on the basis of whether or not the repair facilities can repair a malfunctioning part within the allowable time to prevent delaying any subsequent activity. In this context, a part may be considered "critical" and still not be stocked because, while a shortage of that part may delay one or more activities, it need not affect the overall measure of effectiveness. Actually "critical" means "requires further analysis."

true for parts R and S and demands r and s. The computational techniques used within the computer program to eliminate non-critical parts and demands (Parts Reduction Program) are described in the Appendix.

Reducing the Size of the Network

The network shown in Fig. 10 can be reduced to an equivalent network that contains the first and last nodes (1 and 8) and the demand and fill nodes for demands r and s (2, 3, and 7). First, part T is non-critical, so that, regardless of whether or not demand t occurs, the longest path time between nodes 2 and 7 (via nodes 4 and 6) is 13 days. Thus nodes 4 and 6 can be eliminated, and the five arcs (2-4), (2-5), (4-6), (4-7), and (6-7) can be replaced by the single arc (2-7), with an activity time of 13 days. The longest path time between nodes 3 and 7 is either 8 days (i.e., the repair time for part S) or 6 days (i.e., the nominal schedule), depending on whether or not demand s occurs. Therefore, the two arcs (3-5) and (5-7) can be replaced by a single arc (3-7) with a time of 6 days. Thus, by considering only demand and fill nodes of critical demands, one can reduce the nominal network shown in Fig. 10 to the equivalent network shown in Fig. 11.

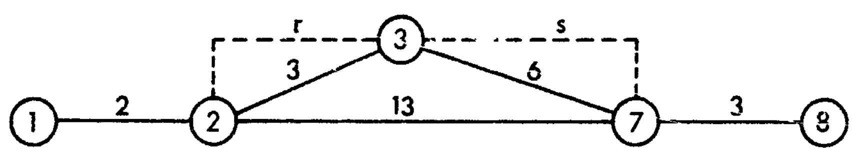


Fig. 11—Equivalent network

The nominal network with 8 nodes and 9 activities is reduced to an equivalent network with 5 nodes and 5 activities. The computational techniques used within the computer program to reduce the size of the network (Network Reduction Program) are described in the Appendix.

III. EXTENSIONS AND LIMITATIONS OF THE NETWORK APPROACH

Section II described a network approach to evaluating and selecting replacement-parts policies. Implicit in the examples were some simplifying assumptions. This section will indicate what the assumptions were, how some of them can be relaxed, and what some of the limitations of the approach are. Problems common to evaluating and selecting a replacement-parts policy will be considered first, and then those problems peculiar to selection.

EVALUATING AND SELECTING A REPLACEMENT-PARTS POLICY

The assumptions in Sec. II pertained to the prelaunch operations, demands for parts, the supply system, and measures of effectiveness.

1) Prelaunch Operations. The schedule of operations can be represented by a project network. In particular, the amount of time required to perform the scheduled activities is not a random variable. Also, the amount of time required to remove and replace a part, given that a replacement is available, is negligible. Finally, the operations pertain to only one vehicle.

2) Demands for Parts. Demands for replacement parts occur at nodes in the network of scheduled operations, and the points at which these demands must be filled are identifiable as nodes in the network. The demands for replacement parts occur independently of one another, and with probabilities that do not depend upon the past history of the parts being replaced. Finally, for any given part, the demands occur in series within the network. That is, the demand and fill nodes for a given part all occur on one path in the network, (e.g., in the example in Sec. II, both demands for part R occur on the path 1-2-3-4-6-7-8. If the fill node for the first demand for part R had been node 5 instead of node 3, then this assumption would have been violated).

3) Supply System. The supply is a two-echelon system in which the first echelon consists of local stocks and the second echelon is local bench repair. A malfunctioning part is put into bench repair as soon as it has been removed from the vehicle. Required repair time for the malfunctioning part is a constant depending only upon the type of part.

4) Measures of Effectiveness. The measure of effectiveness used to evaluate the replacement-parts policy for the example was project time (or delay to the nominal schedule). Thus, in selecting a stocking policy for the assumed repair capability, an effort was made to minimize expected delay.

The ways in which some of these assumptions and restrictions can be removed or alleviated are as follow.

Random Activity Times

Random activity times can be included in the analysis. In the Evaluation and Stock Selection models this means sampling for the times of activities as well as for the occurrence of demands.

Non-Negligible Remove and Replace Times

Remove and replace times can be included in the time required to perform non-scheduled activities -- thereby removing the assumption that they are negligible. When the non-scheduled activity includes repair of the malfunctioning part, the remove and replace times should be added to the repair time. When the non-scheduled activity is obtaining a replacement part from stock, the time associated with this activity should be the remove and replace times for the part. Non-negligible remove and replace times necessitate a modification of the procedure for augmenting the nominal network (i.e., adding arcs to it). Whereas for negligible remove and replace times arcs were added only to represent bench repair activities, now, with non-negligible remove and replace times, one must add arcs to represent the remove and replace activities for every realized demand. Having to remove and replace a part might delay a subsequent activity regardless of whether or not the part is available from stock. Furthermore, one must add the remove and replace times to the repair time to obtain the total time for this activity. Suppose, for example, that Fig. 12 represents a nominal network. The activity times are indicated above each arc. The two dashed lines, c and d, represent two possible demands for the same part. That is, the malfunctioning part discovered at event 2 must be removed and replaced by event 3, and the same process occurs for possible demand d.

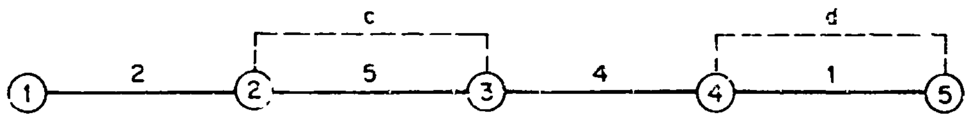


Fig. 12—Nominal network

Figure 13 represents the augmented network when both demands occur and only one of the part is stocked. The time assigned to the added arc (2-3) is the remove time, r_1, plus the replace time, R_1, for the first demand. The same holds for the added arc (4-5). The time assigned to the repair arc (2-5) is the sum of the bench repair time, b, the remove time, r_1, for the first demand, and the replace time, R_2, for the second demand. In this context, retest time can be included as part of the replace time.

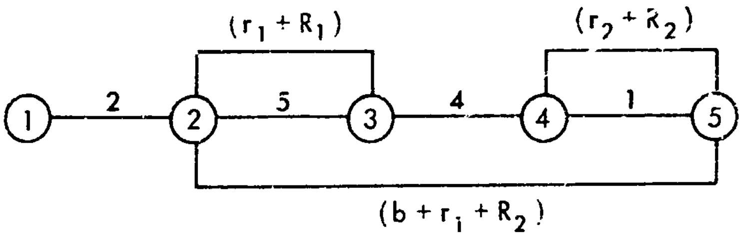


Fig. 13—Augmented network: two demands, one part in stock

This treatment of remove, replace, and retest times is only a partial answer to the problem, since there may be some scheduled activities that are constrained by the remove, replace, and retest activities but not by the activity of obtaining a replacement part. Such constraints are not covered by the above approach.

Multi-Vehicle Projects

Multi-vehicle projects (i.e., concurrent preparation of several vehicles for launch) can be combined into one large network. This approach, however, will result in the occurrence of "parallel" demands for the same part which require special treatment. This problem is discussed below under "Parallel Demands."

Demands Occuring During Activities

It might be desirable, by allowing demands to occur during activities, to weaken the assumption that demands occur only at nodes. However, the effect of this generalization can be obtained by breaking some of the activities into a series of smaller activities and allowing demands to occur at the nodes which have been added.

Statistical Dependence Among Demands

The assumption that demands occur independently probably is not too gross considering the usual unreliability of demand data. However, basically there is no reason why a more complicated joint probability distribution for demands could not be used, although doing so might require considerable change in the sampling techniques described in the Appendix. It is essential to the network approach that the probability of a demand can be specified by knowing only where in the schedule of activities it might occur and not the exact time that it might occur. Therefore, if the probability that a demand occurs at a given node depends upon the time of the node, then that time must be approximated (e.g., use the nominal network to compute the node time).

Parallel Demands

The constraint that the demand and fill nodes for a given part all occur on one path in the network can be relaxed, provided a priority for filling "parallel" demands is specified. A part has "parallel" demands when all of the possible demands for that part do not occur on one path in the network. This problem will occur, for example, with multi-vehicle projects.

Suppose that Fig. 14 represents a nominal network. The activity times are indicated above each arc. The two dashed lines, c and d, represent two possible parallel demands for the same part. That is, the malfunctioning part which is removed at event 3 must be replaced at event 5, the same process being carried out r possible demand d. When

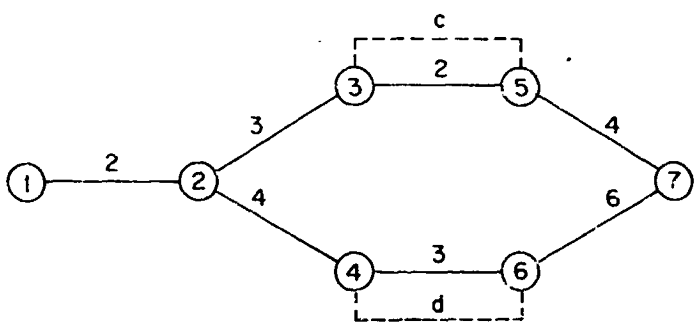
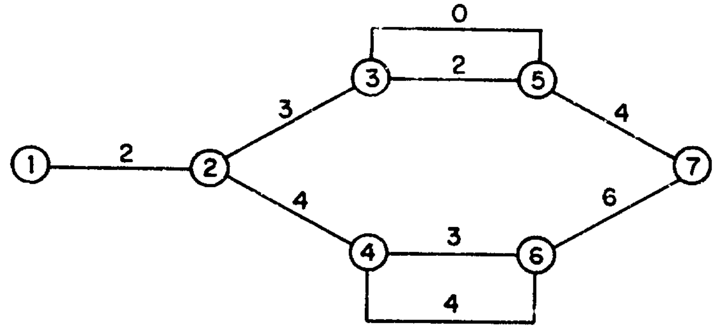


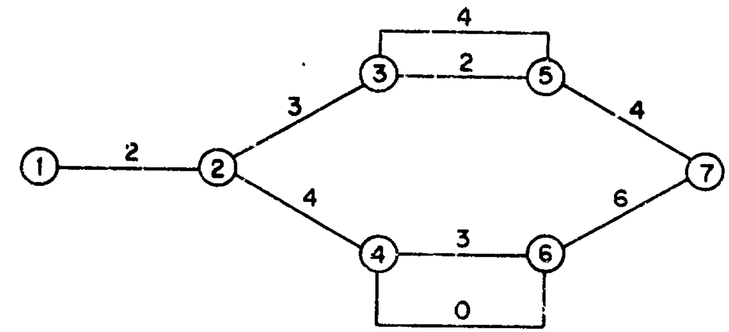
Fig. 14—Nominal network

both demands occur and only one of the part is in stock there are four ways to modify the nominal network as shown in Figs. 15a,b,c, and d. (The repair time for the part is 4 days.)

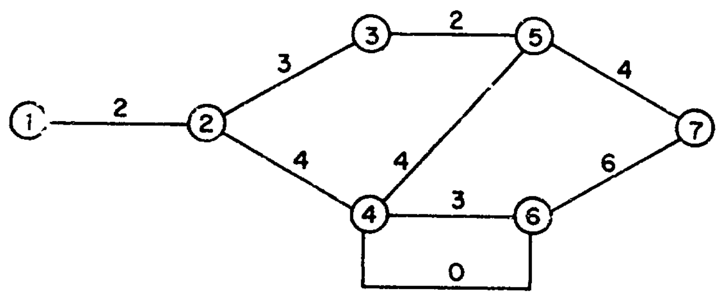
In order first to modify the original network so that it reflects the demands and stock policy and then to compute the earliest time for each event in this augmented network, it is necessary to specify in advance of the time computation which type of modification is to be used. This is, in effect, a specification of the priority system. For the above example, it means that one of the four modifications must be selected <u>before</u> the effect of this choice on the event times is determined. If it is desirable to have a priority system based on event times (e.g., first-come-first-served) then this priority system can be approximated by using event times computed on the basis of the nominal network. This treatment of first-come-first-served would, for the above example, choose the modification shown in Fig. 15d. If this cannot be done then a model that, in effect, simultaneously assigns times to the events and modifies the network, or, in short, a simulation model, is required.



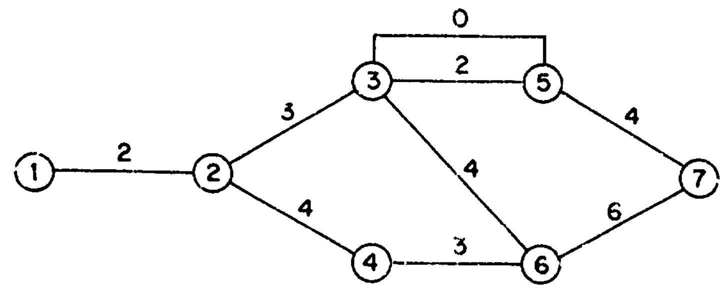
(a) Modification No.1



(b) Modification No.2



(c) Modification No.3



(d) Modification No.4

Fig. 15—Augmented networks

Multi-Echelon Supply Systems

In Sec. II the second echelon was defined as local bench repair, although it could have been interpreted just as easily as bench repair at the manufacturer or reorder from the manufacturer or some other source of resupply. For the purpose of evaluating a replacement-parts policy, the important parameter for a part is the amount of time it takes to obtain a replacement part from the second echelon. This time is defined as the recycle time for the part. For the technique presented here, it is essential that there be only two echelons of supply for each part for any particular evaluation, and that the first echelon be local stocks. However, the second echelon may be different for different parts. To illustrate: for a particular evaluation the second echelon for electronic components might be local bench repair at Merritt Island, and for mechanical components the second echelon might be repair at the manufacturer's main plant.

Different Reorder Policies

In Sec. II, the "reorder" policy was to "order" a part from the second echelon as soon as a part was removed from stock. It is possible to represent different reorder policies. For example, consider the policy which requires that no part be reordered unless there is none of that part in stock and there is a demand for the part. Under such a policy, the representation of replacement activities will be the same as that described in Sec. II, except that each replacement activity will have as its first node the node at which the demand causing it occurred, rather than some earlier demand node. Assuming negligible remove and replace times, the times on the added activities would be zero when there is a replacement in stock and would be equal to the recycle time when a part is not in stock.

Random Recycle Times

It is possible to make the recycle times for parts random variables. In the Stock Selection Model and the Evaluation Model this means that in addition to sampling for the occurrence of demands, one would also sample for recycle times.

Different Measures of Effectiveness

Any function u of the times $t_1, \ldots, t_n$ of the various nodes $1, \ldots, n$ in the network may be used as the measure of effectiveness. For example, if $u(t_1, \ldots, t_n) = -t_n$ and n is the last node in the network, then maximizing E(u), where E is mathematical expectation, is the same as minimizing expected project time. On the other hand, if $u(t_1, \ldots, t_n) = 1$ for $t_n \leq d$ and 0 when $t_n > d$, then E(u) is the probability of completing the project within d days. Suppose the project is a two-vehicle project, and that node i represents the completion of countdown for the first vehicle and node n for the second. If $u(t_1, \ldots, t_n) = 1$ when $t_i \leq d$ and $t_n \leq e$ and 0 otherwise, then E(u) is the probability of launching the first vehicle by time d and the second vehicle by time e.

SELECTING A REPLACEMENT-PARTS POLICY

In addition to the foregoing problems that are common to both evaluating and selecting, there are some that are peculiar to the problem of selecting a replacement-parts policy, i.e., those connected with using marginal analysis and those connected with formulating an adequate cost model.

Interaction Among Parts and Limitations of Marginal Analysis

The network approach allows one to assess the relative criticality of parts not only in context with the schedule of activities and the resupply facilities, but also in relation to the other demands for parts. The approach allows one to assess the extent of the interaction among parts. Two types of interaction exist: 1) the effectiveness of stocking part j increases as the stock level of part i is increased, and 2) the effectiveness of stocking part j decreases as the stock level of part i is increased.

In order to illustrate the first type of interaction, suppose the nominal schedule and possible demands r and s for parts R and S respectively are as represented in Fig. 16. The recycle times for R and S are respectively 5 days and 4 days. If R and S are not stocked, then

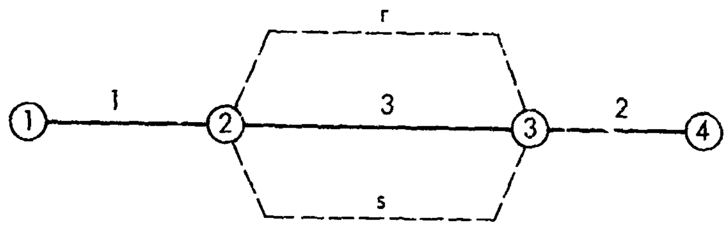


Fig. 16—Nominal schedule and possible demands for R and S

the project time is reduced by stocking S when a demand occurs for S <u>only</u> when no demand occurs for R. However, after R has been stocked, the project time is reduced by stocking S when a demand occurs for S <u>whether or not</u> a demand occurs for R. Thus, the effectiveness of stocking S increases as the stock level of R is increased.

Suppose, to illustrate the second type of interaction, that Fig. 17 represents the nominal network As before, if R is not stocked, then the project time is reduced by stocking S when a demand occurs for S <u>only</u> if a demand occurs for R. But, after stocking part R the project time is <u>never</u> reduced by stocking S. Here, the effectiveness of stocking S decreases as the stock level of R is increased.

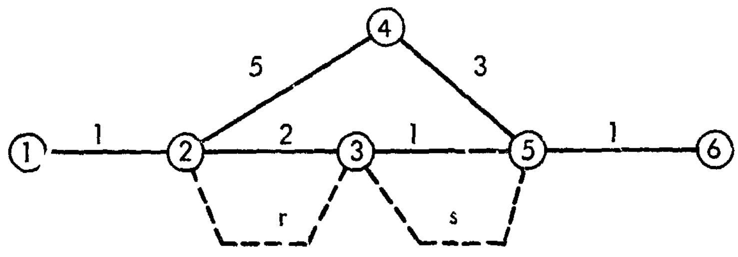


Fig. 17—Nominal network

Because of the interaction among parts, the marginal analysis approach may not always give optimum solutions when it is used to determine stock levels. This approach consists of sequentially adding parts to stock, the part added at each step being the one that yields the greatest ratio of incremental gain in expected effectiveness to incremental cost.

Suppose, for example, that Fig. 18 represents the nominal network and possible demands r, s, and t for three parts R, S, and T, respectively. The recycle times for parts R, S, and T are respectively 3, 3, and 2 days, and the relative costs are respectively 4, 6, and 10 dollars. If one assumes that all three demands always occur, then the marginal analysis approach of looking at one part at a time gives a non-optimum solution, because, for this situation, the incremental effectiveness of stocking one of T is 1 day while the incremental effectiveness of stocking R or S is 0 days. Therefore, one would stock 1 of T at a cost of $10 and never stock R or S. However, it is clear from the example that a better policy is to stock both R and S, rather than T, and obtain an incremental effectiveness of 2 days, rather than 1, for the $10 investment.

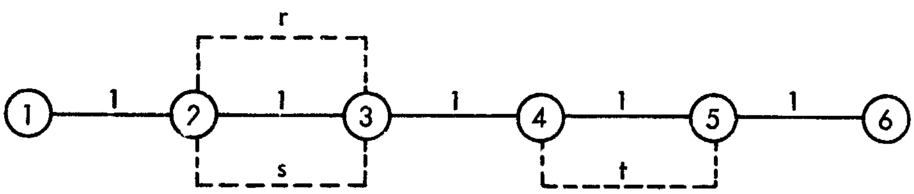


Fig. 18—Nominal network and possible demands for R, S and T

Fortunately, if the probabilities of the demands are low enough, the correct choice will emerge. It will be assumed, for simplicity, that the probability of demand for each part is the same. Let:

p = Probability of demand for the i^{th} part;

Δ_i = Incremental effectiveness for stocking the i^{th} part, when no parts are carried in stock;

c_i = Cost of stocking the i^{th} part; and

E = Mathematical expectation.

Then:

$$E[\Delta_R] = 2p(1-p)$$

$$E[\Delta_S] = 2p(1-p)$$

$$E[\Delta_T] = p$$

Part R would be stocked before part T if:

$$E(\Delta_R)/c_R > E(\Delta_T)/c_T \quad ,$$

or

$$E(\Delta_R)/c_R - E(\Delta_T)/c_T > 0 \quad .$$

Now:

$$E(\Delta_R)/c_R - E(\Delta_T)/c_T = 2p(1-p)/4 - p/10 = p(.8-p)/2 \quad ,$$

so the last term is positive if p is less than .8, in which case part R would be stocked before part T. After part R has been stocked, the expected incremental effectiveness of part S increases, so that it, too, would be stocked before part T, and the optimal stockage would be obtained.

Now, suppose the nominal network and possible demands for the same part R and T are represented by Fig. 19.

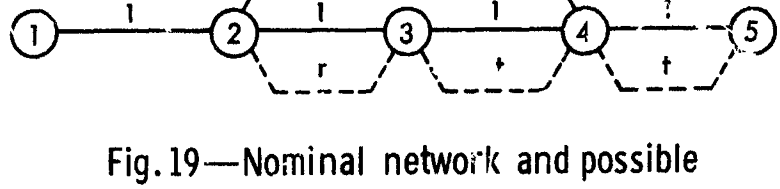


Fig. 19—Nominal network and possible demands for R and T

If one assumes that all three demands always occur, then the incremental effectiveness for stocking R is 1 day and for T is 2 days. But because R costs $4 and T costs $10, the greatest incremental effectiveness for incremental cost is obtained by stocking R. Once R is stocked, the incremental effectiveness for stocking T reduces to 1 day. Assuming it is worthwhile to stock part T, an investment of $14 dollars in parts R and T results in a project time of 6 days. However, it is clear that a better policy is to stock only part T, which will give a project time of 6 days for an investment of only $10. The difficulty with the above marginal analysis is that one cannot "sell back" a part once it has been bought. However, as in the previous example, the correct choice will be made when the probabilities are low enough. For this example, p must be less than 2/3.

Thus, it appears that as the probabilities of the demands decrease the effects of interactions decrease and marginal analysis becomes more valid. In the prelaunch environment, demand probabilities are low; hence the limitations of marginal analysis discussed above are not likely to present a serious problem.

Alternative Cost Models

The procedure being used here to select "good" stock levels for given resupply facilities (i.e., bench repair, manufacturer, etc.) is an application of marginal analysis. The procedure is to generate a sequence of stocking policies by finding the part that, when added to stock, results in the greatest expected incremental effectiveness per incremental cost. The network approach presented here is primarily a technique for evaluating expected effectiveness and not cost. No attempt has been made here to develop new cost models. Implicit in the example discussed in Sec. II is a linear cost model; that is, if Q_i is the quantity of part i in stock, then the cost of the replacement-parts policy is given by

$$a + \sum_i c_i Q_i \quad ,$$

where a and the c_i are fixed parameters that do not depend upon the stocking policy but may depend upon other aspects of the replacement-parts policy such as bench repair facilities. In this case, c_i is what has been called the cost of stocking one of part i. In principle, there is no reason why a more general non-linear cost model cannot be used; however, such a model may lead to problems on the cost side akin to the interaction problems on the effectiveness side, which were discussed above.

CONCLUDING REMARKS

As the foregoing discussion suggests, some of the simplifying assumptions made in Sec. II may be removed without changing the basic approach, which is to add to the network of scheduled activities arcs that represent unscheduled activities. Some of the limitations of the approach have also been indicated. The limitations connected with evaluation stem mainly from one's being unable to characterize completely an unscheduled activity before attaching times to the nodes of the network (e.g., time-dependent priorities for drawing spares). If these limitations are important, then it is necessary to utilize an approach that simultaneously assigns times to the nodes and characterizes unscheduled activities--in short, simulation with its accompanying longer computer running times and more extensive data requirements.

The computational techniques which may be used to implement some of the extensions described above are presented in the Appendix.

IV. NUMERICAL EXAMPLES

The purpose in presenting the following two numerical examples is to illustrate and to demonstrate the feasibility of applying to a life size problem the approach described in Sec. II. Example A is based upon data obtained from an aircraft modification program. As it turns out, Example A is almost trivial. For a more interesting example, the data were modified to constitute Example B. The examples were then used to compare the network approach to two non-network approaches. The computation times cited with the examples are based upon four computer programs: a Parts Reduction Program, a Network Reduction Program, a Stock Selection Model, and an Evaluation Model. The first two programs are written in FORTRAN IV. The last two are written in SIMSCRIPT.* All four programs were run on an IBM 7044 computer. SIMSCRIPT was chosen because it provided an easy method for specifying the dimensions of arrays at run time rather than at compile time. Also, storing the network as a SIMSCRIPT "set" made it easier to program the algorithm for interpreting demands. The execution times for the Evaluation Model and the Stock Selection Model would have been substantially less if they had been written in FORTRAN or machine language. However, since these programs were written as research rather than as production tools SIMSCRIPT was used. Of course, these times could be reduced still further by using a computer faster than the IBM 7044.

EXAMPLE A

Figure 20 is a project network of the schedule for an aircraft modification project. The network consists of 202 arcs (activities) and 115 nodes (events). It was possible to obtain the bench repair time, dollar value, and required demand data for 86 high-value parts, for which there were 99 possible demands.

---

*H. M. Markowitz, B. Hausner, and H. W. Karr, SIMSCRIPT: A Simulation Programming Language, The RAND Corporation, RM-3310-PR, November 1962.

BLANK PAGE

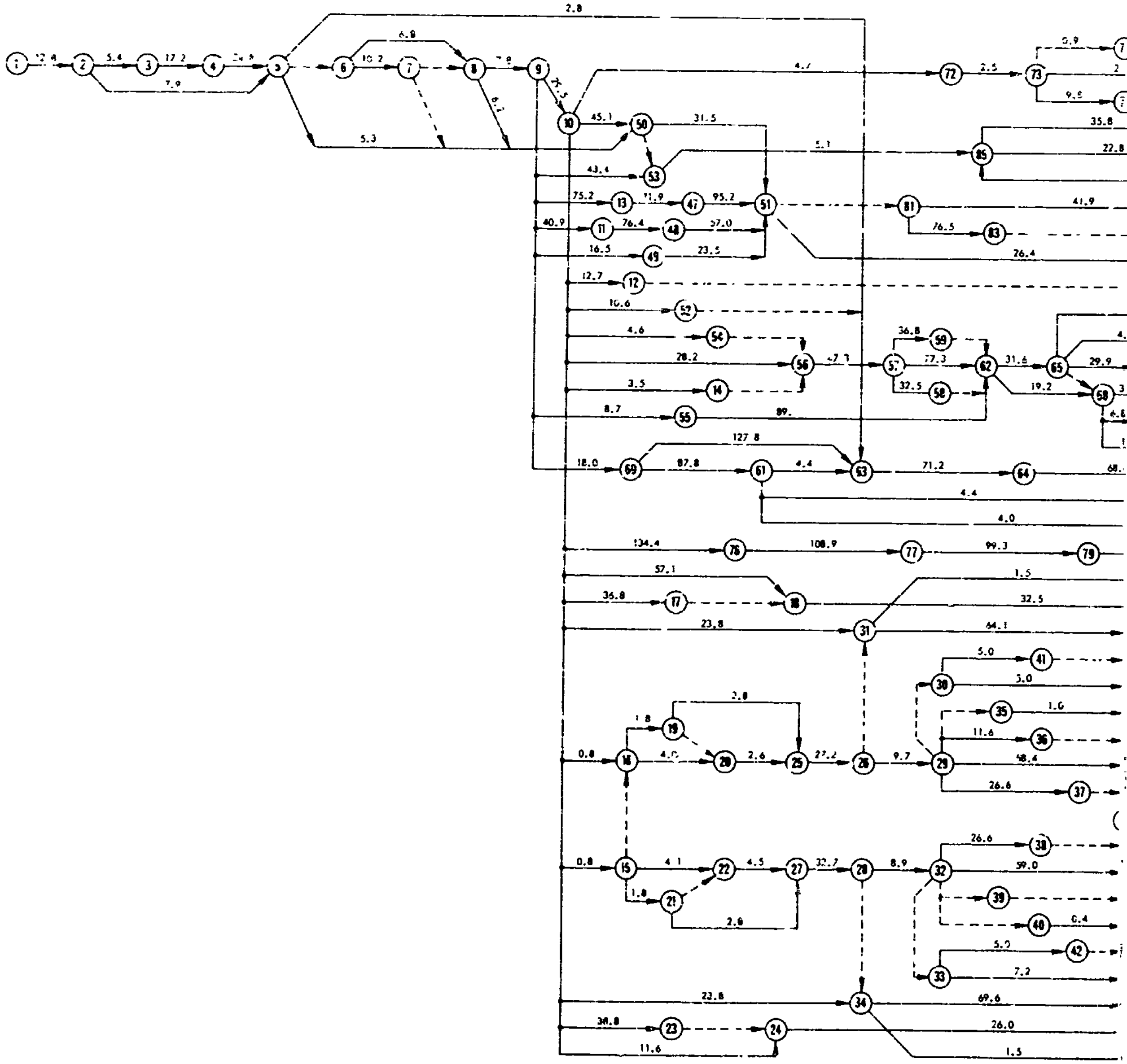
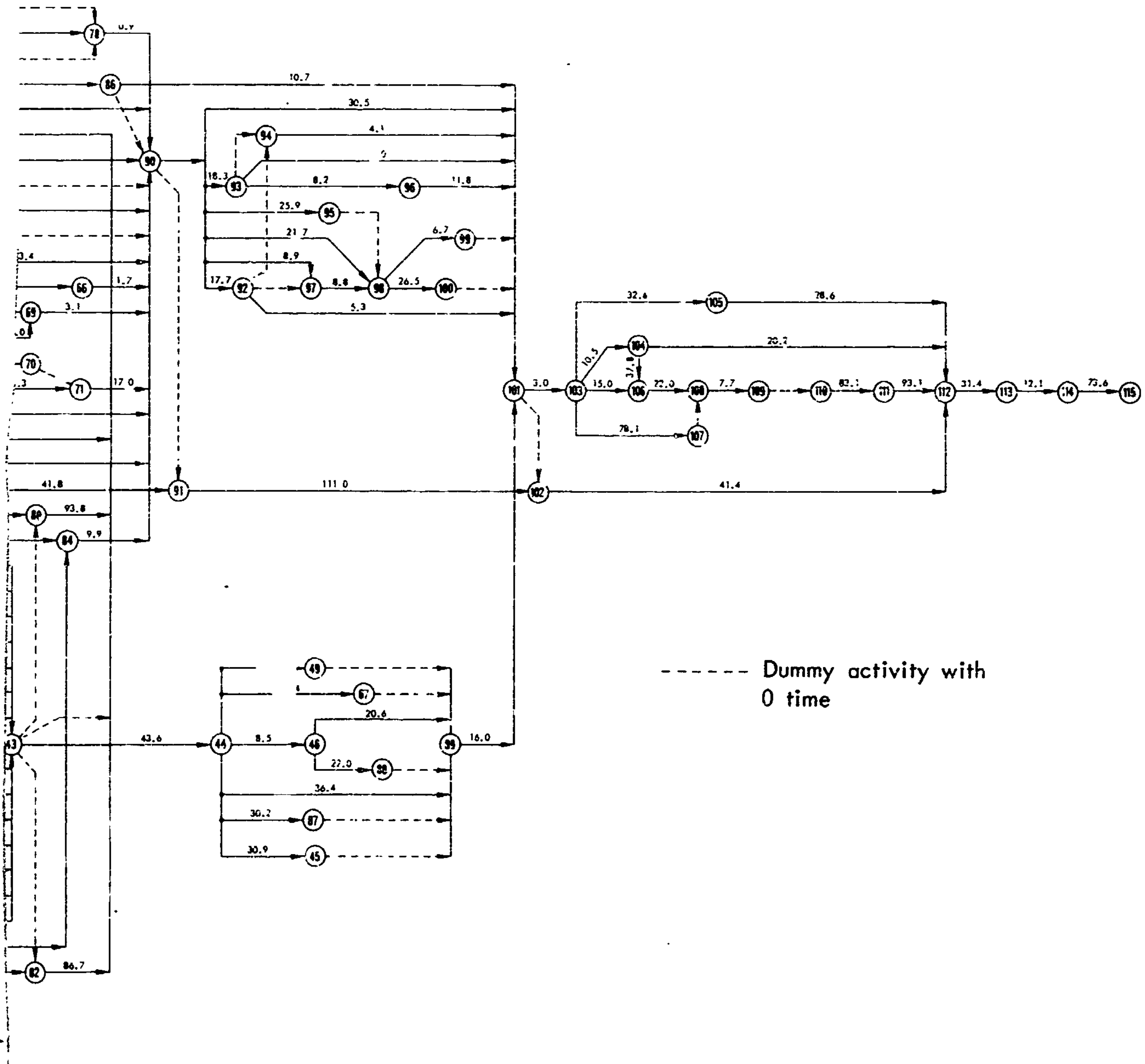


Fig. 20—Nominal networ



k for examples A and B

Eliminating Non-Critical Parts

By comparing the bench repair times with the allowable times for possible demands, 72 non-critical parts were identified. The 14 remaining parts are listed in Table 8. There were 14 possible demands for these parts which are listed in Table 9. Computer time required for parts reduction was 5 seconds.

Reducing the Size of the Network

By considering the 14 possible demands, it was possible to reduce the nominal network of 202 arcs and 115 nodes to an equivalent network of 14 arcs and 11 nodes, as shown in Fig. 21. Computer time required for network reduction was 2 seconds.

Selecting Stock Levels

Project time was used as the measure of effectiveness. Thus the stock selection index was the reduction in average project time per thousand dollars of incremental investment. The indexes for each of the 14 parts were calculated by the Stock Selection Model and are tabulated in Table 10. The first column gives the indexes when no parts are stocked. A zero index indicates that stocking the corresponding part does not reduce project time. In fact, only parts 430 and 440 need to be considered for stock, and 440 is the first candidate. (Actually the difference between the two parts is not significant, so either one is a candidate.) After part 440 has been stocked, its index reduced to zero, and part 430 was the only candidate left (see column 2, Table 10). Thus, the sequence of stock policies, for the given bench repair facilities, consists of stocking nothing, stocking one of part 440, and stocking one of part 440 and one of part 430. There is no need to stock any of the other 84 parts, 12 of which are "critical." Although these 12 parts were "critical" in the sense that a shortage of one of them would delay one or more activities, shortages among these parts otherwise have no effect, evidently, on total project time.

Table 8

CRITICAL PARTS FOR EXAMPLE A

Part Number	Repair Time	Dollar Cost
10	192	3752
120	112	4094
180	192	3759
190	192	3752
340	128	880
350	80	879
430	176	692
440	176	692
460	112	400
480	128	10789
490	128	10789
500	128	9293
840	64	877
850	64	899

Table 9

CRITICAL POSSIBLE DEMANDS FOR EXAMPLE A

Part Number	Demand Node	Fill Node	Proba-bility
10	10	18	.50
120	9	57	.05
180	10	24	.50
190	8	18	.05
340	10	24	.05
350	10	18	.05
430	13	47	.05
440	13	47	.05
460	10	18	.05
480	6	18	.05
490	6	24	.05
500	6	18	.05
840	10	24	.05
850	10	24	.05

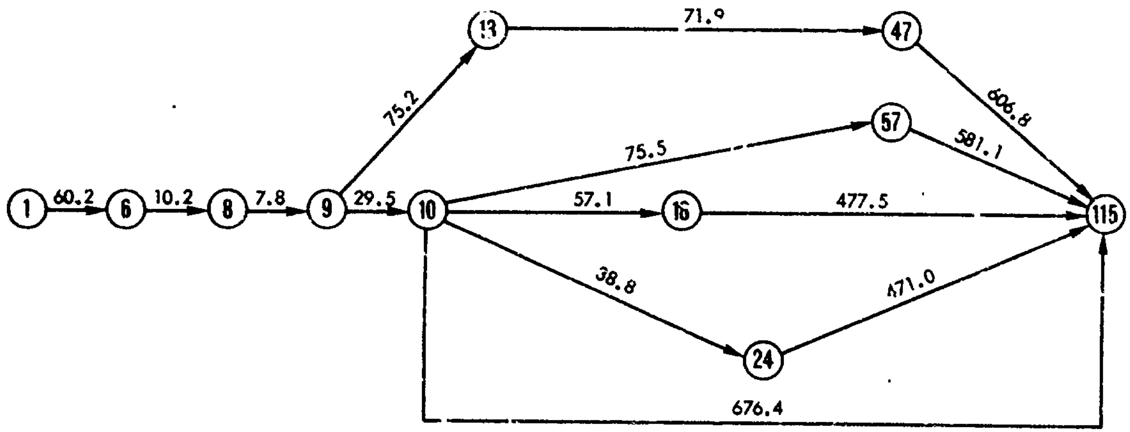


Fig.2ĭ—Equivalent network for example A

Each entry in Table 10 is based on a sample size of 100. The estimated standard error of estimate of each non-zero index was approximately 3 per cent of the index. Total computer time required for stock selection was 2 minutes.

Table 10

STOCK SELECTION INDEXES FOR EXAMPLE A

Part	Part Added to Stock		
	None	440	430
10	0	0	0
120	0	0	0
180	0	0	0
190	0	0	0
340	0	0	0
350	0	0	0
430	6.837	7.013	0
440	7.056	0	0
460	0	0	0
480	0	0	0
490	0	0	0
500	0	0	0
840	0	0	0
850	0	0	0

Evaluating Cost/Effectiveness

The average effectiveness, expected project time, for each of the replacement-parts policies was estimated with the Evaluation Model. The results are tabulated in Table 11 and plotted in Fig. 22. For this example, cost includes only the cost of the part and does not include storage or administration expenses, and the like; nor does it include the investment in repair facilities.

Each of the first two expected project times in Table 11 is based on a sample size of 1000. The third is the nominal project time. The estimated standard errors of the estimates for the first two project times were respectively .85 hours and .67 hours. Each value is based on the same sample of demands. (This was accomplished by starting the random number generator at the same point for each evaluation). As a result of this application of correlated sampling,

Table 11

COST/EFFECTIVENESS FOR EXAMPLE A

Part Added to Stock	Cumulative Cost	Expected Project Time
None	0	841
440	692	837
430	1384	832

any errors in the estimates are likely to be in the same direction. For example, if the estimate from the first evaluation were too high because of a particularly damaging sample, then the same sample is likely to yield an overestimate from the second evaluation. (Thus, if 841 is too high, then 837 is probably too high.) This technique is especially useful for comparing different cost/effectiveness curves, since if one curve is too high (or too low) then the other curves are likely to be also. Computer time required to evaluate cost/effectiveness was 65 seconds.

Discussion

Example A is presented mainly to demonstrate the application of our network approach to establish a replacement-parts provisioning policy. This example serves especially to show how the original problem can be reduced. As it turned out, there is little or no stocking problem for this project -- which is in itself worthwhile information. Actually, such a result is reasonable because the schedule of activities was carefully planned to avoid a spare parts stocking problem. The probabilities for many of these possible demands exceed .50. The investment in repair facilities had already been made; the sequence of activities was therefore scheduled in context with the repair facilities.

EXAMPLE B

In order to present an example that is closer to the space vehicle prelaunch environment, some of the data in Example A were

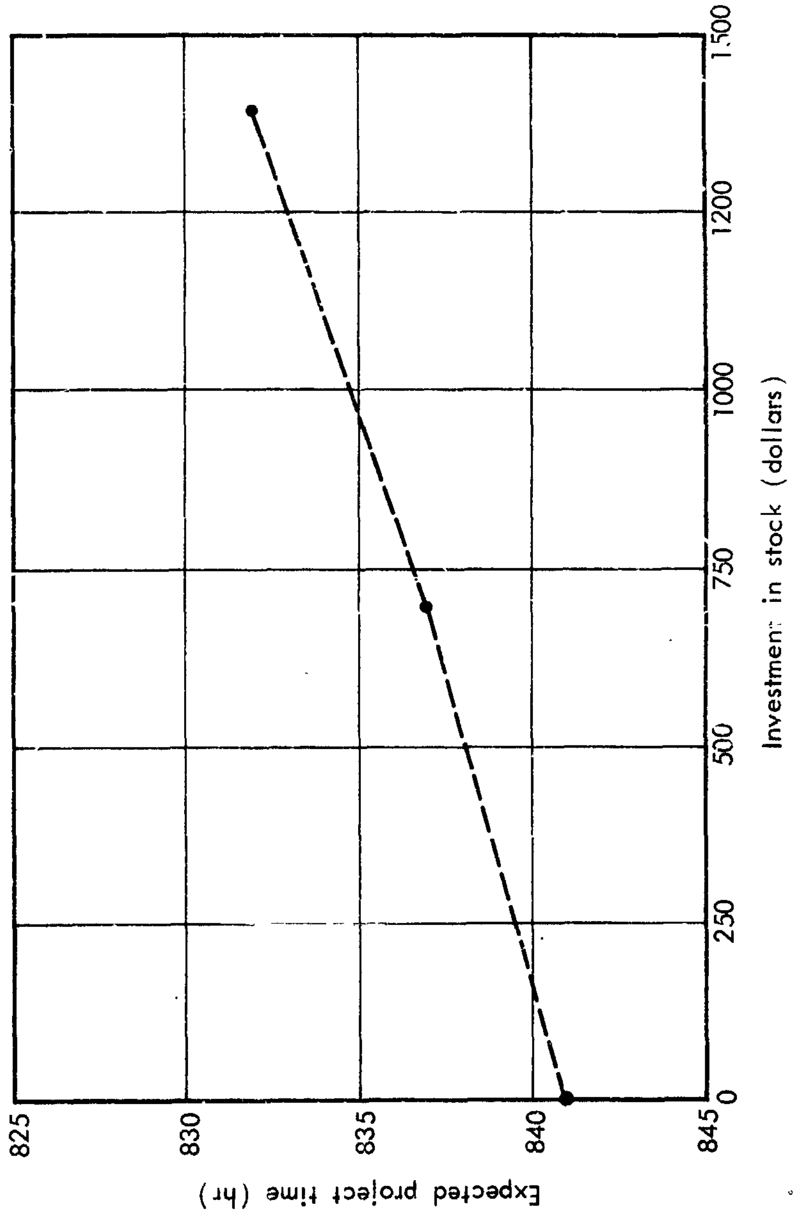


Fig. 22—Cost/effectiveness curve for Example A

altered. First, 20 possible demands were added to increase the number of parts with multiple demands. Next, all of the probabilities were divided by 5 to more nearly approximate the reliabilities found in the space environment. Finally, all of the bench repair times were doubled to increase the number of critical parts and thereby increase the stocking problem. Increasing the repair times can be interpreted as the result of reducing the investment in repair facilities or of relying more on the manufacturer for replacement parts.

Thus modified, Example A becomes Example B. The nominal network is the same as the one for Example A (Fig. 20). There are 86 high-value parts with a total of 119 possible demands.

Eliminating Non-Critical Parts

A comparison of the bench repair times with the allowable times identified 46 non-critical parts; the 40 critical parts are listed in Table 12. The 55 possible demands for these parts are listed in Table 13. Computer time required for parts reduction was 6 seconds.

Reducing the Size of the Network

By considering the 55 possible demands, the nominal network of 202 arcs and 115 nodes was reduced to an equivalent network of 33 arcs and 26 nodes which is shown in Fig. 23. Computer time required for network reduction was 4 seconds.

Selecting Stock Levels

The stock selection indexes, reduction in average project time per thousand dollars of incremental investment for each of the 40 parts, were calculated by the Stock Selection Model. Only those parts with initial indexes greater than zero are tabulated in Table 14. Each column corresponds to a different stock policy. For instance, column 9 represents a stock consisting of one each of parts 10, 20, 180, 430, 440, and 830 and two of part 170. An index listed in column 9 indicates the relative preference for adding one of the part to the stock corresponding to column 9. Thus the part with the

Table 12

CRITICAL PARTS FOR EXAMPLE B

Part Number	Repair Time	Dollar Cost	Part Number	Repair Time	Dollar Cost
10	384	3752	455	224	400
20	96	1406	460	224	400
50	96	830	470	288	3202
90	288	12709	480	256	10789
110	96	830	490	256	10789
120	224	4094	500	256	9293
140	96	898	510	96	10076
170	544	7053	530	96	1191
180	384	3759	570	96	1680
190	384	3752	620	96	2509
210	96	846	660	96	1501
340	256	880	710	96	1898
350	160	879	750	96	1145
380	96	5273	790	672	7854
390	96	5072	800	288	6143
400	96	5378	820	288	7854
410	96	5106	830	256	1228
430	352	692	840	128	877
440	352	692	850	128	899
450	96	4975	860	96	3430

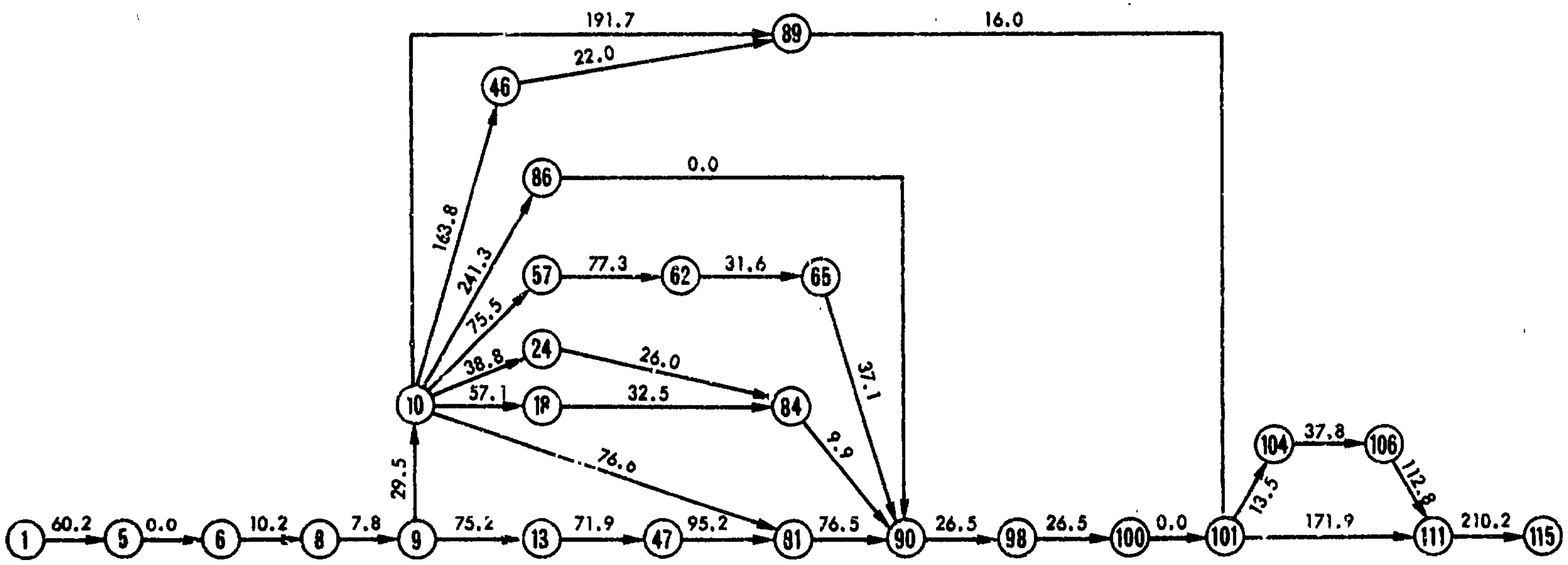


Fig.23—Equivalent network for example B

Table 14

STOCK SELECTION INDEXES FOR EXAMPLE B

Part	Part Added to Stock												
	None	170	20	430	830	440	180	10	170	830	455	170	830
10	1.874	2.792	2.792	2.792	2.891	2.891	3.154	0	0	0	0	0	0
20	3.707	3.707	0	0	0	0	0	0	0	0	0	0	0
90	0.013	--	--	--	0.025	--	--	--	0.036	--	--	--	0.040
120	0.040	--	--	--	0.077	--	--	--	0.111	--	--	--	0.120
140	0.001	--	--	--	0.002	--	--	--	0.002	--	--	--	0.002
170	9.499	2.165	2.165	2.165	2.231	2.532	2.713	2.988	0.354	0.354	0.354	0.016	0.016
180	1.364	2.728	2.728	2.759	2.846	2.881	0	0	0	0	0	0	0
190	0.114	--	--	---	0.204	--	--	--	0.256	0.261	0.261	0.261	0.262
340	0.005	--	--	--	0.031	--	--	--	0.037	--	--	--	0.042
350	0.004	--	--	--	0.006	--	--	--	0.006	--	--	--	0.006
390	0.035	--	--	--	0.038	--	--	--	0.038	--	--	--	0.038
430	2.437	3.292	3.292	0	0	0	0	0	0	0	0	0	0
440	2.250	3.075	3.075	3.084	3.142	0	0	0	0	0	0	0	0
455	0.210	--	--	--	0.476	--	--	--	0.649	0.684	0	0	0
460	0.085	--	--	--	0.234	--	--	--	0.076	0.076	0.076	0.076	0.076
470	0.024	--	--	--	0.052	--	--	--	0.074	--	--	--	0.080
510	0.001	--	--	--	0.003	--	--	--	0	--	--	--	0
530	0.001	--	--	--	0.002	--	--	--	0.002	--	--	--	0.002
620	0.001	--	--	--	0.001	--	--	--	0.001	--	--	--	0.001
750	0.003	--	--	--	0.003	--	--	--	0.003	--	--	--	0.003
790	0.062	--	--	--	0.109	--	--	--	0.150	0.152	0.152	0.157	0.157
800	0.012	--	--	--	0.026	--	--	--	0.040	--	--	--	0.043
820	0.008	--	--	--	0.018	--	--	--	0.027	--	--	--	0.032
830	1.581	3.149	3.149	3.149	1.405	1.405	1.469	1.532	1.532	0.307	0.307	0.307	0.051
860	0.090	--	--	--	0.100	--	--	--	0.099	0.099	0.101	0.101	0.101

Note: -- means these indexes were not computed (see text, page 47).

largest index is added to stock, and new indexes are calculated for all the parts and listed in the next column of the table.

From a practical point of view, the selection process can be made more efficient (through reduction of computer time) by calculating indexes for a limited number of parts. For instance, after calculating the indexes for all parts (first column), it was decided to consider only those parts with indexes larger than 1.0. However, because of the possible interactions among parts, it is prudent periodically to check for the extent of this interaction by calculating the indexes for all parts. Also, once the index for a part reduces to zero, it will remain zero for all subsequent stocking policies.

Each non-zero index was estimated with a sample size of 100. For most of the selection indexes the estimated standard error of estimate was less than 10 per cent of the estimate of the index. The samples from row to row in Table 14 were independent, thereby allowing one to make statistical comparisons among indexes for different parts. However, within any row the sample used was the same; thus any change in the index for part i when part j is added to stock represents the effect of interaction between parts i and j, rather than sampling error. Total computer time required to generate Table 14 was 17.5 minutes.

<u>Evaluating Cost/Effectiveness</u>

The expected effectiveness (in this case, expected project time) for each of the replacement-parts policies was evaluated with the Evaluation Model. The results are tabulated in Table 15 and plotted in Fig. 24. As in Example A, cost includes only the cost of the part.

Only every other expected project time, beginning with the first, was estimated with the Evaluation Model. A sample size of 1000 was used for each evaluation. The intervening values were computed by using the estimates of incremental effectiveness that were obtained as a byproduct of the stock selection process. The standard error of estimate decreased from 2.4 hours for the first expected project time in Table 15 to .5 hours for the last. Total computer time required for the seven evaluations was 10 minutes.

Table 15

COST/EFFECTIVENESS FOR EXAMPLE B

Part Added to Stock	Cumulative Cost	Expected Project Time
None	0	966
170	7053	896
20	8459	891
430	9151	889
830	10379	886
440	11071	884
180	14830	875
10	18582	863
170	25635	842
830	26863	841
455	27263	840
170	34316	837
830	35544	836

Discussion

Example B demonstrates the application of marginal analysis to generate a sequence of stocking policies for given repair capabilities. The selection process was made by recalculating indexes for only a limited number of parts, thereby reducing computer time. Just how many parts should be considered at a time is subject to good judgment. We chose to start the analysis by recomputing only those indexes that had an initial value greater than 1.0 -- which appeared to be a reasonable value (see column 1 of Table 14). As parts were stocked, the original list of 7 candidates was reduced, as were their indexes. Finally the decision to add more candidates to the list shifted attention to those parts with indexes greater than .095 (see column 10). The analysis was terminated at column 13. Note that the selection indexes in column 2 are generally higher than those in column 1, indicating that the effect of demands for these parts was hidden by the demands for part 170. However, the ranking of the selection indexes is not altered appreciably. On the other hand, the demands for part 20 do not appear to influence the effect of demands for other parts (see column 3).

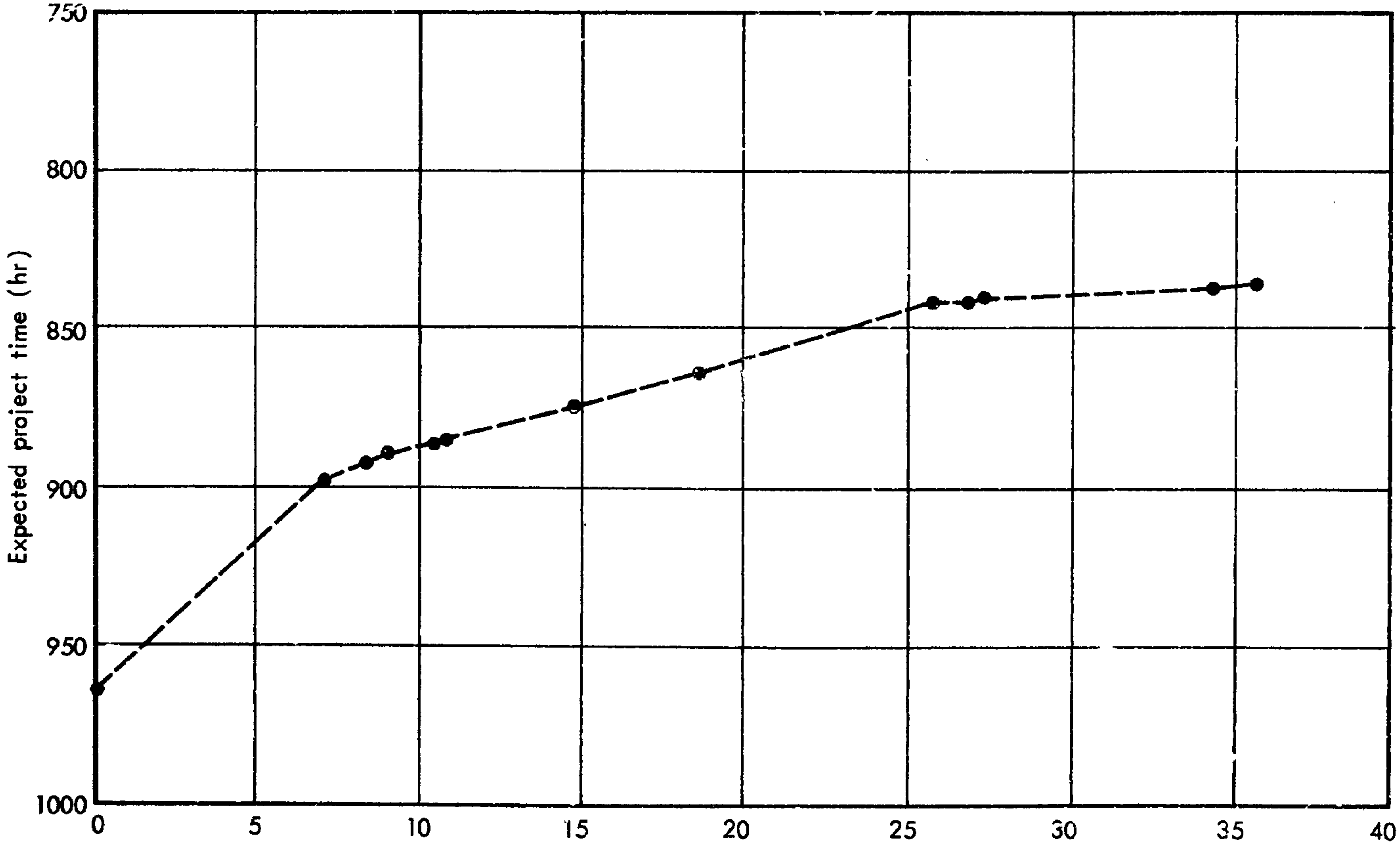


Fig. 24—Cost/effectiveness curve for Example B

COMPARISON WITH TWO NON-NETWORK APPROACHES

For comparison of non-network approaches with the network approach expected shortages were used instead of expected delay to the project as an alternative criterion for selecting stocks. A shortage was defined in one of the following two ways.

> Method I. A shortage occurs when a malfunctioning part is identified at a demand node and no replacement part is available at that demand node. The schedule of activities is virtually ignored because only the demand nodes are considered.

> Method II. A shortage occurs when a malfunctioning part is identified at a demand node and no replacement part is available at the fill node. Here, the schedule of activities is partially considered because a distinction is made between demand nodes and fill nodes.

In both Methods I and II, a malfunctioning part can be removed, repaired, and returned to stock. In order to determine if such a stock replenishment prevents a shortage, the times of the demand and fill nodes are set equal to their nominal times. Actually, Method II is a form of a network approach because it does take into consideration the "partial predictability of demands" in that the distinction is made between demand nodes and fill nodes; to some extent, this accounts for a portion of the "complicated relation that exists between shortages and effectiveness." In particular, the "non-critical" parts are weeded out as potential candidates for stock.

For Example A, a sequence of stocking policies according to both Methods I and II was selected. From the previous analysis, the only parts which, when stocked, will reduce expected project time are parts 430 and 440. As it turns out, Method I results in an investment of $64,810 in 33 unnecessary parts before either parts 430 or 440 are stocked. The corresponding figures for Method II are $7,911 for 3 unnecessary parts. The total cost of parts 430 and 440 is $1,384.

Table 16 tabulates the sequence of stocking policies for Example B using Methods I, II, and III, where III is the network method. Only policies resulting in investments less than $40,000 are given. The unnecessary parts are indicated. Figure 25 shows the cost/effectiveness curves for Method I (curve I), Method II (curve II), and the Network Method (curve III). Each of the evaluations given in Table 16 was based upon the same sample, thereby minimizing the effect of sampling error on comparisons of the three methods. Note that even the crude network approach (Method II) shows a substantial improvement over Method I. However, the Network Method is an improvement over Method II.

Table 16

COST/EFFECTIVENESS FOR METHODS I, II, AND III

EXAMPLE B

Method I			Method II			Method III		
Part Added to Stock	Cumulative Cost	Project Time	Part Added to Stock	Cumulative Cost	Project Time	Part Added to Stock	Cumulative Cost	Project Time
None	0	966	None	0	966	None	0	966
830	1228	964	830	1228	964	170	7053	896
20	2634	958	830	2456	963	20	8459	891
830	3862	957	170	9509	892	430	9151	889
250[a]	4956	957	20	10915	886	830	10579	886
280[a]	6050	957	455	11315	886	440	11071	884
330[a]	7729	957	460	11715	885	180	14830	875
370[a]	9466	957	830	12943	885	10	18582	863
170	16519	886	10	16695	873	170	25635	842
750	17664	886	180	20454	862	830	26863	841
300[a]	18539	886	750	21599	862	455	27263	840
310[a]	19414	886	170	28652	843	170	34316	837
320[a]	20305	886	430	29344	842	830	35544	836
290[a]	21206	886	440[b]	30036	840			
260[a]	22120	886	710[b]	31934	840			
420[a]	23039	886	840[b]	32811	840			
40[a]	23959	886	350	33690	840			
640[a]	26148	886	340	34570	840			
760[a]	27203	886	850[b]	35469	840			
710[b]	29191	886	50[b]	36299	840			
455	29591	886	110[b]	37129	840			
460	29991	886	210[b]	37975	840			
830	31219	885	140	38873	840			
10	34971	873						
180	38730	862						

[a]These parts are non-critical, hence stocking them does not reduce expected project time.

[b]Stocking these parts does not reduce expected project time.

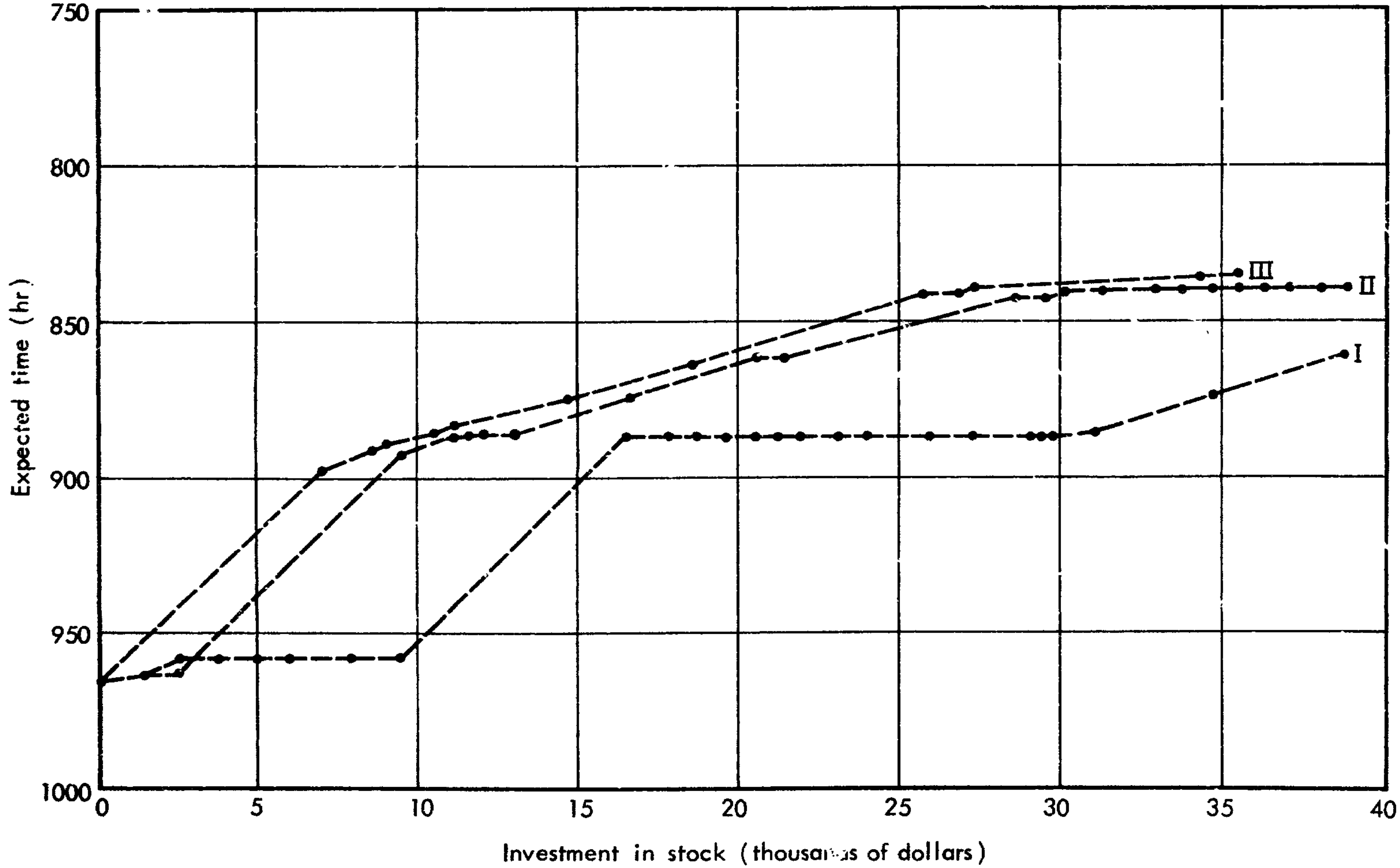


Fig. 25 — Cost/effectiveness curve for Example B: methods I, II, III

V. CONCLUDING REMARKS

The objective of this Memorandum has been to present methods to assist in selecting and evaluating replacement-parts policies for Apollo prelaunch operations -- methods that are complex enough to include the special features of the problem discussed in Sec. I, but are simple enough to have feasible applications. The approach described in Sec. II takes into account these special features: the partial predictability of demands and the complicated relationship between parts shortages and effectiveness. Section III has pointed out how the basic approach can be extended. In Sec. IV, the network approach was applied to an aircraft modification project, and the means of selecting a stocking policy was demonstrated.

The required data inputs for the basic computer models are:

1) A description of the schedule of activities for the project in network form with the associated activity times.
2) A list of parts with their recycle times and costs.
3) A list of possible demands for each part, with demand nodes, fill nodes, and probabilities of occurrence.

The procedure is to:

1) Eliminate non-critical parts.
2) Reduce the size of the network.
3) Select a sequence of stock levels for a given resupply capability.
4) Evaluate the expected measures of effectiveness for the stocking policies established in 3) above.
5) Repeat 3) and 4) for as many different resupply capabilities as desired.
6) Construct the cost/effectiveness curves for the various replacement-parts policies.

The network approach fills a gap that exists between analytical inventory models that do not include the special features discussed in Sec. I and elaborate simulation models that have greater data requirements and longer computer running times. The salient feature of the network approach is that replacement-parts policies are evaluated and stocking policies are selected by evaluating the supply system in terms of its effect on the overall project, rather than using some measure that looks at the supply in isolation from the project. This effect is measured in terms of the amount of delay to the project which may be expressed in a variety of ways (e.g., expected project time or probability of meeting the launch window or windows for a multivehicle project).

APPENDIX

INTRODUCTION

Section II of this Memorandum has described an approach to the problem of determining a parts policy for a project. Now computational techniques that may be used to implement this approach will be described. At appropriate points, there will be indications as to how these techniques may be modified to implement the extensions described in Sec. III. The approach consists of selecting a sequence of stock policies (specifications of how much of each part should be stocked) for a given backup resupply capability (a specification of recycle time for each part). Each stock policy in the sequence, together with the resupply capability, has a cost and an effectiveness. On the basis of these data a cost/effectiveness curve may be drawn which reflects the opportunities available under the given resupply capability. One such curve may be drawn for each resupply capability under consideration. Having done this, one can make a simultaneous choice of resupply capability and stock policy by selecting a point on one of the curves.

The remainder of this Appendix is devoted to describing techniques for selecting and evaluating a stock policy for a fixed resupply capability. In order to do this, one needs data on the schedule of operations, the parts, and the possible demands for these parts. The data on the schedule of operations should be in project network form, i.e., a list of arcs (each representing an activity), the identification of the beginning and ending node for each arc, and the time required for the activity represented by the arc. For each part one needs to know its recycle time and its cost. The demand data should be in the form of a list of possible demands, where each possible demand is characterized by the part which might be demanded, by its demand node (node in the network at which the demand occurs), by its fill node (node in the network by which point the demand must be filled), and by the probability that the demand will occur. These data may then be used as inputs for two Monte Carlo models: a Stock Selection Model used to determine the sequence of stock policies, and an Evaluation Model used to evaluate the stock policy together with the fixed resupply capability. Before starting the Monte Carlo computations, however, it is possible to perform preliminary computations that reduce the size of the problem. These techniques are described first. The method for selecting a sequence of stock policies is described next. Finally, the evaluation of a replacement-parts policy is described.

REDUCING THE SIZE OF THE PROBLEM

As Sec. II indicates, there are two ways to reduce the size of the problem. One can reduce the number of parts and demands to be considered, and one can reduce the size of the network.

Reducing the Number of Parts and Demands

Suppose that for a given possible demand there is a path from its demand node to its fill node that is longer than the recycle time for the part. In this case, the malfunctioning part can be removed at the demand node, repaired (or reordered), and replaced at the fill node without delaying any of the activities in the project. If this is the case for all possible demands for a given part, there is no need to stock the part, so that part and all possible demands for it may be eliminated from further consideration. Thus, a reduction in the list of parts may be accomplished as follows. First for each possible demand, compute its "allowable time," i.e., the length of the longest path in the nominal network from its demand node to its fill node. Next, eliminate from the list of parts any part whose recycle time is no more than the smallest "allowable time" for all possible demands for the part. If remove and replace times are not negligible, then the allowable time should be compared with recycle time plus remove and replace times. Finally, do away with all possible demands for eliminated parts.

To compute the "allowable time" for a possible demand, one needs an algorithm for computing the longest path from one node in a project network to another. The algorithm described here is a slight modification of an algorithm for computing the longest path in a project network. For a given arc α, let α^1 denote its first node and α^2 denote its second node. Assume the arcs $\alpha_1, \alpha_2, \ldots, \alpha_s$ of the network are indexed in such a way that $\alpha_i^2 = \alpha_j^1$ implies $i < j$. (Since project networks are acyclic, this is always possible.) Let $\tau(\alpha)$ denote the "length" of α, i.e., the time required by the activity represented by the arc α.

Let m be an arbitrary node in the network. The following algorithm will set t(n) for each node n in the network, equal to the length of the longest path from m to n, if there is such a path; it will set $t(n) = -\infty$ if there is no such path.

Step 1: Set $t(n) = -\infty$ for each node n.

Step 2: Set $t(m) = 0$; set $i = 1$.

Step 3: Set $t(\alpha_i^2) = \max\,[t(\alpha_i^2),\ t(\alpha_i^1) + \tau(\alpha_i)]$.

Step 4: If $i = s$ stop; otherwise set $i = i + 1$ and go to Step 3.

<u>Reducing the Size of the Network</u>

As exemplified in Sec. II, and described in more detail below, the effect on the project of the occurrence of demands for parts is reflected entirely in terms of the addition of arcs to the network. Each added arc represents an activity that is needed to fill a demand. Whenever an arc is added, its first node is the demand node for some demand, and its second node is the fill node for some demand. In assessing the effect of demands for parts on the project, one is interested only in the times at selected nodes of the network (e.g., the last node -- see Sec. III under "Different Measures of Effectiveness."). The time at a node is the length of the longest path from the first node of the network to the node in question. A node will be called "special" if it is either one of the "selected nodes," the first node of the network, or a demand or fill node for some possible demand. The longest path from the first node in the network to any special node may be broken into shorter paths, each of which begins at a special node, ends at a special node, and has no special node as an intermediate node. Now if one augments the network by adding arcs each of whose end points is a demand or a fill node, the only effect on the above longest path will be to replace some of the shorter paths into which it has been broken by some of the arcs that have been added. For this reason, the nominal network may be replaced by one whose only nodes are the special nodes, and whose arcs are constructed as follows. For each pair of special nodes m and n, determine whether there is a path in the nominal network that goes from m to n, has no special nodes as intermediate nodes, and is at least as long as any other path from m to n in the nominal network. If there is such a path, construct an arc whose time is equal to the length of the path and whose first node is m and last node is n.

The algorithm presented below will determine for a given special node m, and all other nodes n, whether or not there should be an arc from m to n in the reduced network, and if so, how long that arc should be. As above, it is assumed that the arcs $\alpha_1, \ldots, \alpha_s$ have been indexed in such a way that $\alpha_i^2 = \alpha_j^1$ implies $i < j$, where, as before, α^1 is the first node of the arc α, and α^2 is the second node. Again, let $\tau(\alpha)$ be the time associated with the arc α. The algorithm makes use of a function f defined on the nodes of the nominal network by: $f(n) = 1$ if $n = m$ or n is not a special node, and $f(n) = 0$ if n is a special node and $n \neq m$. The algorithm computes two functions g and t of the nodes that have the following interpretation: if $(1 - f(n))g(n) = 1$ then there should be an arc from m to n in the reduced network whose time is $t(n)$; otherwise there should not be an arc from m to n in the reduced network. The algorithm is:

Step 1: Let $i = 1$, $g(m) = 1$, $g(n) = 0$, for $n \neq m$, $t(m) = 0$, and $t(n) = -\infty$, for $n \neq m$.

Step 2: If $t(\alpha_i^2) < t(\alpha_i^1) + \tau(\alpha_i)$, go to Step 3.

If $t(\alpha_i^2) = t(\alpha_i^1) + \tau(\alpha_i)$, go to Step 4.

If $t(\alpha_i^2) > t(\alpha_i^1) + \tau(\alpha_i)$, go to Step 5.

Step 3: Set $t(\alpha_i^2) = t(\alpha_i^1) + \tau(\alpha_i)$.

Set $g(\alpha_i^2) = g(\alpha_i^1)\ f(\alpha_i^1)$.

Go to Step 5.

Step 4: Set $g(\alpha_i^2) = g(\alpha_i^2)\ g(\alpha_i^1)\ f(\alpha_i^1)$.

Step 5: If $i = s$, stop; otherwise, set $i = i + 1$ and go to Step 2.

SELECTING A SEQUENCE OF STOCK POLICIES

Ideally, one would like to generate a sequence of _efficient_ stock policies: an "efficient stock policy" is a policy that is more effective than any policy which costs less, and at least as effective as any policy that costs the same. Unfortunately, the method about to be described cannot be guaranteed to do this. In fact (see Sec. III) it is possible to construct examples where the method selects inefficient stocks. However, such examples depend upon a relatively high occurrence of particular combinations of demands. In the context of prelaunch operations, where individual demand probabilities are likely to be low, such combinations have a much lower probability. For this reason, the method described should result in _nearly_ efficient stock policies. The procedure is as follows. Let the first policy in the sequence consist of carrying no parts in stock. Then find a part that when added to stock results in the greatest incremental expected effectiveness per incremental cost. This stock represents the second policy in the sequence. Subsequent policies are obtained by repeating the process.*

---

*Let $A(r_1 \ldots, r_\mu)$ be the average effectiveness when stocking r_i [illegible] part i, $i = 1, \ldots, \mu$. Let

$$[\text{illegible}](r_1, \ldots, r_\mu) [\text{illegible}] (r_1, \ldots r_{i-1}, r_i+1, r_{i+1}, \ldots, r_\mu) - A(r_1, \ldots, r_\mu).$$

[illegible] $\ldots r_\mu)$ depends only on r_i, for each $i = 1, \ldots, \mu$, then A

For application of the above procedure, a method is needed to determine the average incremental expected effectiveness obtained by adding a part to stock. A Monte Carlo computer program (Stock Selection Model) has been written for this purpose. In broad outline one iteration of the program runs as follows. First, a set of demands is generated on the basis of the possible demands and their probabilities that are input to the program. If activity times are to be random variables, their times may be generated at this point. On the basis of these demands and the parts that are carried in stock, arcs that represent activities needed to fill the demands are added to the nominal network. Times for the nodes of the augmented network are then computed, and a measure of effectiveness u_0 is calculated. Then the quantity of the part being analyzed is increased by one, the demands are reinterpreted, the node times for the new network are recomputed, and a measure of effectiveness, u_1, is computed. The difference $u_1 - u_0$ represents the incremental effectiveness of increasing by one the quantity stocked of the part being analyzed for the particular demands that were generated. This increment is then averaged in with the increments obtained on previous iterations. The process is repeated for as many iterations as desired.

---

is called separable. If A were separable and in addition $\Delta_i A(r_1, \ldots, r_\mu)$ were a non-increasing function of r_i, then the above procedure would lead to efficient stock policies. (See, for example, C. T. Whitehead, Selection of Spares and Redundancy for the Apollo Spacecraft, The RAND Corporation, RM-4177-NASA, August 1964.) Unfortunately, A is not separable in this problem. However, the dependency of $\Delta_i A(r_1, \ldots, r_\mu)$ on r_j, $j \neq i$, occurs because of the possibility of simultaneous demands for parts i and j. Since the probability of such simultaneous demands is small in comparison with the probabilities of the individual demands, it seems reasonable that the dependency of $\Delta_i A(r_1, \ldots, r_\mu)$ on r_j, $j \neq i$, should be small in relation to its dependency on r_i, and that the stock selection procedure should lead to nearly efficient stocks.

Thus the computations in the model fall into four main classes: 1) those required for sampling to see which possible demands are actually realized; 2) those required to interpret the demands by adding arcs to the nominal network; 3) the computation of node times for the augmented network; and 4) the computation of the estimates of expected incremental effectiveness (and also standard deviation of incremental effectiveness, and standard error of estimate of expected incremental effectiveness).

Sampling for Demands

When estimating the expected incremental effectiveness obtained by adding one of part P to stock, the sample is stratified on the basis of the number of demands for part P since this incremental effectiveness is highly correlated with the number of demands for part P. Therefore, by stratifying in this way, one obtains better estimates for smaller sample sizes. Suppose that there are m possible demands for part P, and n possible demands for the other parts. Define random variables $X_1, \ldots, X_m$ and $Y_1, \ldots, Y_n$ by:

$$X_i = \begin{cases} 1, & \text{if the } i^{th} \text{ demand for part P occurs;} \\ 0 & \text{otherwise.} \end{cases}$$

$$\Pr(X_i = 1) = p_i, \quad \text{for } i = 1, \ldots, m.$$

$$Y_j = \begin{cases} 1, & \text{if the } j^{th} \text{ demand for parts other than P occurs;} \\ 0 & \text{otherwise.} \end{cases}$$

$$\Pr(Y_j = 1) = q_j, \quad \text{for } j = 1, \ldots, n.$$

It is assumed that the X_i and Y_j are independent.

Since there are m possible demands for part P, there are m + 1 strata, where the k^{th} stratum, for k = 0, 1, ..., m, is defined by the restriction

$$(1) \qquad \sum_1^m X_i = k$$

on the values of the random variables $X_1, \ldots, X_m$.

The sampling problem divides into three others: 1) how to determine the probability of the k^{th} stratum; 2) how to determine the sample size for the k^{th} stratum; and 3) how to sample from the k^{th} stratum.

Determining the probability of the k^{th} stratum. The problem is to find $\Pr(\sum_1^m X_i = k)$ for each $k = 0, \ldots, m$. This may be done by means of a recursion. For this purpose, let

$$C_{\ell k} = \Pr(\sum_1^\ell X_i = k), \quad (\ell = 1, \ldots, m;\ k = 0, \ldots, \ell).$$

Since the X_i are independent and $\Pr(X_i = 1) = p_i$, one has, for $0 < k < \ell$,

$$\Pr(\sum_1^\ell X_i = k) = \Pr(X_\ell = 1 \ \&\ \sum_1^{\ell-1} X_i = k-1) + \Pr(X_\ell = 0 \ \&\ \sum_1^{\ell-1} X_i = k)$$

$$= p_\ell \Pr(\sum_1^{\ell-1} X_i = k-1) + (1 - p_\ell)\Pr(\sum_1^{\ell-1} X_i = k).$$

Thus,

$$(2) \qquad C_{\ell k} = p_\ell C_{\ell-1,k-1} + (1-p_\ell)C_{\ell-1,k}, \quad (0 < k < \ell).$$

For $\ell = k > 1$, one has

$$\Pr(\sum_1^\ell X_i = \ell) = \Pr(X_\ell = 1 \ \&\ \sum_1^{\ell-1} X_i = \ell-1)$$

$$= p_\ell \Pr(\sum_1^{\ell-1} X_i = \ell-1).$$

So

$$(3) \qquad C_{\ell\ell} = p_\ell C_{\ell-1,\ell-1}, \quad (\ell > 1).$$

For $\ell > 1$ and $k = 0$, one has

$$\Pr(\sum_1^\ell X_i = 0) = \Pr(X_\ell = 0 \ \&\ \sum_1^{\ell-1} X_i = 0) = (1-p_\ell)\Pr(\sum_1^{\ell-1} X_i = 0).$$

Thus,

(4) $$C_{\ell 0} = (1-p_\ell)C_{\ell-1,0}\ , \quad (\ell > 1)\ .$$

Finally, $Pr(X_1 = 0) = 1-p_1$ and $Pr(X_1 = 1) = p_1$, so

(5) $$C_{10} = 1-p_1\ , \quad C_{11} = p_1\ .$$

Equations 2 through 5 may be used to compute the values of $Pr(\sum_1^m X_i = k) = C_{mk}$, i.e., to compute the probabilities of the k^{th} stratum, for $k = 0, \ldots, m$. An algorithm that does this is given below. When the algorithm is completed, C_k will be $Pr(\sum_1^m X_i = k)$, i.e., the probability of the k^{th} stratum for $k = 0, \ldots, m$.

Step 1: Let $C_0 = 1 - p_1$, $C_1 = p_1$, and $\ell = 1$.

Step 2: If $\ell = m$, stop; otherwise set $\ell = \ell+1$ and continue.

Step 3: Set $C_\ell = p_\ell C_{\ell-1}$ and $k = \ell-1$.

Step 4: Set $C_k = p_\ell C_{k-1} + (1-p_\ell)C_k$.

Step 5: If $k > 1$, set $k = k-1$ and go to Step 4; otherwise continue.

Step 6: Set $C_0 = (1-p_\ell)C_0$, and go to Step 2.

Determining the sample size for the k^{th} stratum. Suppose that Q is the quantity of part P stocked. The purpose of the Stock Selection Model is to estimate the expected increment in effectiveness that would be obtained were Q increased by one. It is clear that when the number of demands for part P is less than or equal to Q, this incremental effectiveness will be zero. Thus for $k \le Q$, the sample size for the k^{th} stratum can be set to zero. For the remaining strata, $k > Q$, the situation is not so clear. If the standard deviation of the incremental effectiveness in the k^{th} stratum were known, then the sample size for that stratum could be set proportional to the standard deviation multiplied by the probability of the stratum, thereby minimizing the standard error of the estimate for any fixed sample size.* Unfortunately, these standard deviations for $k > Q$ are not known (for $k \leq Q$ they are zero), and there is very little, a priori, that can be said about them. As a compromise, the sample size for the k^{th} stratum, $k > Q$, is set proportional only to its probability, calculated as above.

Sampling from the k^{th} stratum. Since the Y_j are independent of each other and of the X_i, and Eq. 1 does not restrict the values of the Y_j, one may sample for the Y_j independently of each other and independently of the X_i. There is a simple method of doing this. For each $j = 1, \ldots, n$, generate a random number v_j uniformly distributed on the unit interval. (Hereafter any of these will be referred to as a "random number.") Then let $Y_j = 1$, if $v_j < q_j$ and $Y_j = 0$ otherwise. Sampling for values of the X_i is not so simple, since given Eq. 1 they are not independent of each other.

A method is needed that will sample for values $x_1, \ldots, x_m$ of $X_1, \ldots, X_m$ in such a way that the probability of obtaining $x_1, \ldots, x_m$ as values is given by

$$(6) \qquad \Pr(X_1 = x_1 \;\&\; \ldots \;\&\; X_m = x_m | \sum_1^m X_i = k).$$

The method described here accomplishes this as follows. Sample for the first X_i in the sequence $X_1, \ldots, X_m$, for which $X_i = 1$. Given the first i for which $X_i = 1$, sample for the second i for which $X_i = 1$. Continue this process until k of the X_i have been set equal to 1

To carry out the above procedure, one needs the probability that X_{i_β} is the β^{th} 1 in the sequence $X_1, \ldots, X_m$; given that $\sum_1^m X_j = k$;

*See, for example, Herman Kahn, Applications of Monte Carlo, The RAND Corporation, RM-1237-AEC, April 1956, pp. 107, 108.

that X_{i_1} is the first 1 in the sequence; that X_{i_2} is the second 1 in the sequence, ...; and that $X_{i_{\beta-1}}$ is the $\beta-1^{th}$ 1 in the sequence, for any $0 < i_1 < \ldots < i_{\beta-1} < i_\beta \leq m$.

Denote this conditional probability by Q. Then

(7) $$Q = \Pr(X_{i_\beta} = 1 \ \& \sum_{i_{\beta-1}<j\leq i_\beta} X_j = 1 | X_{i_1} = \ldots = X_{i_{\beta-1}} = 1 \ \& \sum_{0<j\leq i_{\beta-1}} X_j = \beta-1 \ \& \sum_1^m X_j = k).$$

Working on the "given" in Eq. 7, one has

(8) $$Q = \Pr(X_{i_\beta} = 1 \ \& \sum_{i_{\beta-1}<j\leq i_\beta} X_j = 1 | [X_{i_1} = \ldots = X_{i_{\beta-1}} = 1 \ \& \sum_{0<j\leq i_{\beta-1}} X_j = \beta-1] \ \& \sum_{i_{\beta-1}<j} X_j = k-\beta+1).$$

From the independence of $X_1, \ldots, X_m$ it follows that the conditions within brackets in Eq. 8 may be removed, so Eq. 8 becomes

(9) $$Q = \Pr(X_{i_\beta} = 1 \ \& \sum_{i_{\beta-1}<j\leq i_\beta} X_j = 1 | \sum_{i_{\beta-1}<j} X_j = k-\beta+1).$$

Thus, Q does not depend upon $i_1, \ldots, i_{\beta-2}$, but only upon $i_{\beta-1}$, i_β and $k-\beta+1$. In fact, Q is simply the probability that X_{i_β} is the first 1 following $X_{i_{\beta-1}}$, given that exactly $k-\beta+1$ ones follow $X_{i_{\beta-1}}$. This probability will be denoted by $D(i_\beta, i_{\beta-1}, k-\beta+1)$, i.e.,

(10) $$D(i,\ell,h) = \Pr(X_i = \sum_{\ell<j\leq i} X_j = 1 | \sum_{\ell<j} X_j = h), \quad (i > \ell \geq 0, \ h \leq m-i).$$

$D(i,0,h)$ is the probability that X_i is the first 1 in $X_1, \ldots, X_m$, given that there are h ones in all.

A method must still be found, however, for computing $D(i,\ell,h)$. Applying the definition of conditional probability to Eq. 10, one has

$$D(i, \ell, h) = \frac{\Pr(X_i = \sum_{\ell<j\le i} X_j = 1 \ \& \ \sum_{\ell<j} X_j = h)}{\Pr(\sum_{\ell<j} X_j = h)}$$

$$= \frac{\Pr([X_i = \sum_{\ell<j\le i} X_j = 1] \ \& \ [\sum_{i<j} X_j = h-1])}{\Pr(\sum_{\ell<j} X_j = h)}.$$

The bracketed conditions in the last expression are independent, so

$$D(i,\ell,h) = \Pr(X_i = \sum_{\ell<j\le i} X_j = 1)\left[\frac{\Pr(\sum_{i<j} X_j = h-1)}{\Pr(\sum_{\ell<j} X_j = h)}\right].$$

The first factor above is simply $p_i \prod_{\ell<j<i} (1 - p_j)$ for $i>\ell+1$, and p_i for $i = \ell+1$. Thus if $E(\ell,h)$ is defined by

(11) $$E(\ell,h) = \Pr(\sum_{\ell<j} X_j = h), \qquad (0 \le \ell < m,\ 0 \le h \le m-\ell),$$

and $F(\ell,i)$ by

(12) $$F(\ell,i) = \begin{cases} p_i, & (0 < \ell+1 = i \le m) \\ p_i \prod_{\ell<j<i} (1-p_j), & (0 < \ell+1 < i \le m) \end{cases} ;$$

then

(13) $$D(i,\ell,h) = F(\ell,i) \left[\frac{E(i,h-1)}{E(\ell,h)}\right].$$

When $\ell = 0$, $E(\ell,h)$ is simply the probabili_ f the h^{th} stratum, whose computation was described above. For $\ell >$ and $m-\ell \geq h > 0$, one has

$$E(\ell-1, h) = Pr(\sum_{\ell-1<j} X_j = h)$$

$$= Pr(X_\ell = 1)Pr(\sum_{\ell<j} X_j = h-1) + Pr(X_\ell = 0)Pr(\sum_{\ell<j} X_j = h)$$

$$= p_\ell E(\ell,h-1) + (1-p_\ell)E(\ell,h).$$

Thus,

(14) $$E(\ell,h) = [E(\ell-1,h) - p_\ell E(\ell,h-1)]/(1-p_\ell), \quad (\ell > 0,\ m-\ell \geq h > 0).$$

For $\ell > 0$ and $h = 0$

$$E(\ell-1, 0) = Pr(\sum_{\ell-1<j} X_j = 0)$$

$$= Pr(X_\ell = 0)Pr(\sum_{\ell<j} X_j = 0)$$

$$= (1-p_\ell)E(\ell, 0),$$

so

(15) $$E(\ell, 0) = E(\ell-1, 0)/(1-p_\ell), \quad (\ell > 0).$$

Finally, as noted above,

(16) $$E(0, h) = Pr(\sum_{1}^{m} X_j = h), \quad (0 \leq h \leq m),$$

whose computation has been described under "Determining the probability of the k^{th} stratum." From Eq. 12, $F(\ell,i)$ obviously satisfies the recursion,

$$(17) \qquad F(\ell,i) = [p_i(1-p_{i-1})/p_{i-1}]F(\ell,i-1), \quad (0 < \ell+1 < i \le m),$$

with $F(\ell,\ell+1)$ given by

$$(18) \qquad F(\ell,\ell+1) = p_{\ell+1}, \quad (0 \le \ell < m).$$

Equations 13 through 18 can now be put together in the form of an algorithm for sampling from the k^{th} stratum. The idea behind the algorithm is to sample for the first i, i_1, for which $X_i = 1$, choosing i_1 with probability $D(i_1,0,k)$. One then samples for the second i, i_2, for which $X_i = 1$, choosing i_2 with probability $D(i_2,i_1,k-1)$. In general, having chosen the first β subscripts, $i_1 < \ldots < i_\beta$, for which $X_{i_1} = \ldots = X_{i_\beta} = 1$, one chooses the $(\beta+1)^{th}$ such subscript $i_{\beta+1}$ with probability $D(i_{\beta+1},i_\beta,k-\beta)$. This choice is made by generating a random number v and setting $i_{\beta+1}$ equal to the first i for which $\sum_{j=i_\beta+1}^{i} D(j,i_\beta,k-\beta) \ge v$. In the algorithm, h is the number of X_j that remain to be set equal to 1, and i is the index of the X_j that is currently being set equal to 0 or 1. Steps 3, 4, and 5 and steps 10, 11, and 12 use Eq. 15 and Eq. 14 to compute $E(i,j)$ and store the result as $E(j)$. Steps 2 and 9 use Eq. 18 and Eq. 17 to compute $F(\ell,i)$, where ℓ is the last j for which X_j has been set equal to 1, and store the result as F. Step 2 also sets A equal to $E(\ell,h)$. Steps 6 and 12 use Eq. 13 to compute $\sum_{j=\ell+1}^{i} D(j,\ell,h)$ and store the result as B.

Step 1: Set $E(j) = Pr\left(\sum_1^m X_\gamma = j\right)$ for $j = 0, \ldots, k$;

set $h = k$; set $i = 1$.

Step 2: Set $F = p_i$; set $A = E(h)$.

Step 3: Set $E(0) = E(0)/(1-p_i)$; set $j = 1$.

Step 4: Set $E(j) = [E(j) - p_iE(j-1)]/(1-p_i)$.

Step 5: If $j < h-1$, set $j = j+1$ and go to Step 4; otherwise continue.

Step 6: Set $B = FE(h-1)/A$, and generate a random number v.

Step 7: If $v \leq B$, set $X_i = 1$ and go to Step 13; otherwise set $X_i = 0$ and continue.

Step:8: Set $i = i+1$.

Step 9: Set $F = [p_i(1-p_i)/p_{i-1}]F$.

Step 10: Set $E(0) = E(0)/(1-p_i)$; set $j = 1$.

Step 11: Set $E(j) = [E(j) - p_iE(j-1)]/(1-p_i)$.

Step 12: If $j < h-1$, set $j = j+1$ and go to Step 11; otherwise set $B = B+FE(h-1)/A$ and go to Step 7.

Step 13: Set $h = h-1$.

Step 14: If $h > 0$, set $i = i+1$ and go to Step 2; otherwise continue.

Step 15: Set $X_j = 0$ for all $j > i$ and stop.

Interpreting the Demands

For each possible demand δ for a given part P, let

δ^d = the demand node of δ

δ^f = the fill node of δ, and

$$X(\delta) = \begin{cases} 1, & \text{if the demand occurs;} \\ 0 & \text{otherwise.} \end{cases}$$

A scheme for setting the values of X has been described above. Here it is assumed that the values of X have already been determined, and a description follows of how the δ^f and δ^d, together with the quantity Q stocked of part P, may be used to interpret the values of X in terms of additional arcs in the network. After a description of the algorithm, its validity will be discussed. If $Q \geq \sum_{\delta} X(\delta)$, then do not add any arcs to the network. Otherwise, assume that the possible demands have been indexed $\delta_1, \ldots, \delta_m$. Let b be the recycle time for part P.

Step 1: Set i = 1.

Step 2: If $X(\delta_i) = 0$, go to Step 8; otherwise continue.

Step 3: Set j = i and k = 0.

Step 4: If k = Q, go to Step 7; otherwise continue.

Step 5: If j = m, stop; otherwise set j = j+1 and continue.

Step 6: If $X(\delta_j) = 1$, set k = k+1 and go to Step 4; otherwise go to Step 5.

Step 7: Add an arc to the network whose first node is δ_i^d, whose second node is δ_j^f, and whose time is b.

Step 8: If i = m-Q, stop; otherwise set i = i+1 and go to Step 2.

The above algorithm provides a valid interpretation of the pattern of demands for P, when the part removed as a result of the i^{th} such demand is recycled and used to fill the $(i+Q)^{th}$ such demand. Implicit in this statement, however, is the assumption that the possible demands for P can be ordered $\delta_1, \ldots, \delta_m$ in a reasonable fashion. Such an ordering is automatically provided when the fill nodes of the possible demands for a given part all lie on a common path in the network. (The difficulties that appear when this condition is not satisfied have been discussed in Sec. II.) Even when this condition is not met, it is sometimes still possible to obtain a reasonable ordering. For example, the possible demands for a part could be ordered on the basis of the latest times, in the nominal network, of their fill nodes. (The latest time of a node is the total project time minus the length of the longest path from that node to the last node in the network.) An ordering of this type would tend to give highest priority to those demands that, in order to avoid project delay, should be filled first.

The algorithm also assumes that remove and replace times are negligible. They can be included, however, as follows. With the notation as above, let $r(\delta)$ be the time required to remove the malfunctioning part when demand δ occurs and let $R(\delta)$ be the time required to replace the part when demand δ occurs. The above algorithm may then be used with Step 7 modified in this way:

Step 7': Add an arc to the network whose first node is δ_i^d, whose second node is δ_j^f, and whose time is $b + r(\delta_i) + R(\delta_j)$.

When $Q > 0$, it is also necessary to perform the following algorithm:

Step 1: Set $i = 1$.

Step 2: If $X(\delta_i) = 0$ go to Step 4; otherwise continue.

Step 3: Add an arc to the network whose first node is δ_i^d, whose second node is δ_i^f, and whose time is $r(\delta_i) + R(\delta_i)$.

Step 4: If $i = m$, stop. Otherwise set $i = i+1$ and go to Step 2.

Random recycle times may be included by sampling for them at Step 7 in the original algorithm, or at Step 7' in the revised algorithm.

Evaluating the Network

The computations required to assign earliest times to the nodes of the network are ordinary project network computations. As before, let α^1 and α^2 be the first and second nodes, respectively, of the arc α. Let $\tau(\alpha)$ be the time required by the activity represented by α, and assume that the arcs $\alpha_1, \ldots, \alpha_s$ have been indexed in such a way that $\alpha_i^2 = \alpha_j^1$ implies $i < j$. The algorithm attaches a time $t(n)$ to each node n in the network; $t(n)$ is equal to the length of the longest path from the first node in the network to n.

Step 1: Set $t(n) = 0$ for each node n, and set $i = 1$.

Step 2: Set $t(\alpha_i^2) = \max[t(\alpha_i^2), t(\alpha_i^1) + \tau(\alpha_i)]$.

Step 3: If $i = s$, stop; otherwise set $i = i+1$ and go to Step 2.

Estimating the Expected Incremental Effectiveness

As pointed out in Sec. III, the measure of effectiveness can be any function of the node times. The incremental effectiveness of adding one of part P to stock is simply the difference between effectiveness before and after making the addition. Let Δ be the incremental effectiveness. Δ is, of course, a random variable since it depends upon which of the possible demands actually occur. Let $\Delta_j^{(k)}$ be the j^{th} observation on Δ from the k^{th} stratum. An estimate, $\widetilde{E(\Delta)}$, of the expectation $E(\Delta)$ is provided by

$$\widetilde{E(\Delta)} = \sum_{k=Q+1}^{m} (C_k/N_k)\left(\sum_{j=1}^{N_k} \Delta_j^{(k)}\right), \tag{19}$$

where C_k is the probability of the k^{th} stratum, N_k is the sample size of the k^{th} stratum, Q is the quantity of part P stocked (before the addition to stock), and m is the number of possible demands for part P. Since sample means are unbiased and consistent estimates of the population mean, this estimate is unbiased and consistent.

The above estimate has a variance $\mathrm{Var}[\widetilde{E(\Delta)}]$ given by

$$\mathrm{Var}[\widetilde{E(\Delta)}] = \sum_{k=Q+1}^{m} [C_k^2 \mathrm{Var}(\Delta^{(k)})/N_k],$$

wh_re $\Delta^{(k)}$ is the restriction of Δ to the k^{th} stratum, and $\mathrm{Var}(\Delta^{(k)})$ is the variance of $\Delta^{(k)}$. Now

$$(20) \qquad \widetilde{\mathrm{Var}(\Delta^{(k)})} = \frac{1}{N_k - 1} \sum_{j=1}^{N_k} (\Delta_j^{(k)})^2 - \frac{1}{N_k^2 - N_k} [\sum_{j=1}^{N_k} (\Delta_j^{(k)})]^2$$

provides an unbiased, consistent estimate of $\mathrm{Var}(\Delta^{(k)})$; hence

$$(21) \qquad \widetilde{\mathrm{Var}[\widetilde{E(\Delta)}]} = \sum_{k=Q+1}^{m} C_k^2 \widetilde{\mathrm{Var}(\Delta^{(k)})}/N_k$$

is an unbiased consistent estimate of $\mathrm{Var}[\widetilde{E(\Delta)}]$.

Since $\widetilde{E(\Delta)}$ is a consistent estimate of $E(\Delta)$, $[\widetilde{E(\Delta)}]^2$ is a consistent estimate of $[E(\Delta)]^2$. Also,

$$(22) \qquad \widetilde{E(\Delta^2)} = \sum_{k=Q+1}^{m} \left(C_k/N_k\right) \sum_{j=1}^{N_k} (\Delta_j^{(k)})^2$$

is a consistent estimate of $E(\Delta^2)$. Thus $\widetilde{E(\Delta^2)} - [\widetilde{E(\Delta)}]^2$ is a consistent estimate of $\mathrm{Var}(\Delta) = E(\Delta^2) - [E(\Delta)]^2$. But it is easily found that this estimate has a downward bias, and the amount of the bias is $\mathrm{Var}[\widetilde{E(\Delta)}]$. Thus

$$(23) \qquad \widetilde{\mathrm{Var}(\Delta)} = \widetilde{E(\Delta^2)} - [\widetilde{E(\Delta)}]^2 + \widetilde{\mathrm{Var}[\widetilde{E(\Delta)}]}.$$

is an unbiased consistent estimate of $\mathrm{Var}(\Delta)$.

In order to compute the estimates 19, 21, and 23, it is not necessary to store all the $\Delta_j^{(k)}$, but only certain cumulative summaries.

In going from one observation to another within the k^{th} stratum, one need only build the cumulative sums $\sum_j \Delta_j^{(k)}$ and $\sum_j (\Delta_j^{(k)})^2$. When all the observations from the k^{th} stratum have been completed, one may compute $\widehat{Var(\Delta^{(k)})}$ according to Eq. 20. In going from stratum to stratum, one need only build the cumulative sums

$$\sum_k (C_k/N_k)(\sum_{j=1}^{N_k} \Delta_j^{(k)});$$

$$\sum_k (C_k^2/N_k)\ \widehat{Var(\Delta^{(k)})};\ \text{and}$$

$$\sum_k (C_k/N_k) \sum_{j=1}^{N_k} (\Delta_j^{(k)})^2 .$$

When all the strata have been covered, these sums may be used in Eqs. 19, 21, 22, and 23 to obtain the estimates $\widehat{E(\Delta)}$, $Var[\widehat{E(\Delta)}]$, and $\widehat{Var(\Delta)}$.

EVALUATING A REPLACEMENT-PARTS POLICY

An Evaluation Model has been developed for evaluating a replacement-parts policy. The detailed computations within the model are essentially the same as in the Stock Selection Model described in the previous section, but overall organization is somewhat different. One iteration of the model proceeds as follows. First, a set of demands is generated on the basis of the possible demands and their probabilities. On the basis of these demands and the parts that are carried in stock, arcs that are needed to fill the demands are added to the nominal network. Times for the nodes of the network are computed; then the measure of effectiveness is computed as a function of these times. This measure is averaged with the measures obtained on previous iterations. The sequence is repeated for as many iterations as desired.

The demands are generated as in the Stock Selection Model, except that the stratification is on the basis of the number of demands for **all** parts, rather than for a single part. The addition of arcs to the network, computation of node times, and computation of the measure of effectiveness are performed exactly as in the Stock Selection Model. The combining of the individual observations on the measure of effectiveness to arrive at estimates of the expected effectiveness, the standard deviation of the effectiveness, and the standard error of estimate of expected effectiveness is exactly as in the Stock Selection Model, except that the observations being combined are observations on effectiveness rather than on incremental effectiveness.

CONCLUDING REMARKS

The algorithms described in this section have been put in the form of four computer programs for the IBM 7044: a Parts Reduction Program, a Network Reduction Program, a Stock Selection Model, and an Evaluation Model.

The Parts Reduction Program has as input the nominal network (a list of activities, with their first and second nodes and times) and a list of possible demands (each characterized by its demand node and fill node). The output is the "allowable" time for each possible demand. The program is written in FORTRAN IV.* A network consisting of 202 arcs and 115 nodes and a list of 119 possible demands required approximately 6 seconds of execution time.

The Network Reduction Program has as input the nominal network, and a list of "special" nodes. The output is a reduced network that has as its nodes only the first and last nodes of the nominal network plus the "special" nodes. This program, too, is written in FORTRAN IV.* A network consisting of 202 arcs and 115 nodes and 24 "special" nodes required approximately 4 seconds of execution time.

The Stock Selection Model has as input the data, described on page 56, on the schedule of operations (nominal network in either reduced or original form), the parts, and the possible demands for them, plus the initial stock level for each part, a specification of the parts to be analyzed, and the sample size for each analysis. The output for each part analyzed is estimates of the part's expected

*However, the subset of FORTRAN IV used is also a subset of FORTRAN II.

incremental effectiveness per incremental cost (and its standard error of estimate), the part's expected incremental effectiveness (and its standard error of estimate), and the standard deviation of the incremental effectiveness. The program is written in SIMSCRIPT. SIMSCRIPT was chosen because it provided an easy method for specifying the dimensions of arrays at run time rather than at compile time. Also, storing the network as a SIMSCRIPT "set" made it easier to program the algorithm for interpreting demands. For Example B given in Sec. V, the analysis of each part required approximately 8 seconds execution time for a sample size of 100. The reduced network for this example consisted of 26 nodes and 33 arcs; there were (after parts reduction) 55 possible demands.

The Evaluation Model has the same input requirements as the Stock Selection Model with respect to the schedule, parts, and possible demands, except that the cost of the parts is not required. In addition, a list of the stocking policies (i.e., a specification of the quantity of each part stocked) to be evaluated and the sample size for each evaluation are required. The outputs for each evaluation are estimates of expected effectiveness, standard deviation of effectiveness, and standard error of estimate of expected effectiveness. The program is written in SIMSCRIPT for reasons the same as those given above. For the previously cited example and for sample sizes of 1000, each evaluation required approximately 90 seconds. The execution times for the Evaluation Model and the Stock Selection Model would have been substantially less if they had been written in FORTRAN or machine language; however, since these programs were written as research rather than as production tools SIMSCRIPT was used.

The computer programs were written for the simplified version of the network approach; however, Sec. III has indicated how many of these simplifying assumptions can be removed, and this Appendix has suggested what the effect will be on the computations. This Appendix has not discussed the specific changes required in the computations to incorporate multi-echelons, different reorder policies, different probability structures, or different cost models. Each of these extensions to the basic approach has many variations, and, to the extent that they can be incorporated in the computer programs, it is best to do so in the context of a specific application.