AD-A102 513   UNIVERSITY OF SOUTHERN CALIFORNIA REDONDO BEACH BEHAV--ETC   F/G 9/2
DESIGN FOR THE MAINTAINER: PROJECTING MAINTENANCE PERFORMANCE F--ETC(U)
JUL 81   D M TOWNE, M R FEHLING, N A BOND   N00014-80-C-0493

UNCLASSIFIED   TR-95   NL

1 OF 2
AD A
02513

DESIGN FOR THE MAINTAINER:

PROJECTING MAINTENANCE PERFORMANCE FROM DESIGN CHARACTERISTICS

Douglas M. Towne
Michael R. Fehling
Nicholas A. Bond

With the assistance of
Anthony K. Mason
Mark C. Johnson

July 1981

BEHAVIORAL TECHNOLOGY LABORATORIES

Department of Psychology

University of Southern California

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

DESIGN FOR THE MAINTAINER:

PROJECTING MAINTENANCE PERFORMANCE FROM DESIGN CHARACTERISTICS

Douglas M. Towne
Michael R. Fehling
Nicholas A. Bond

With the Assistance of
Anthony K. Mason
Mark C. Johnson

July 1981

Technical Report No. 95

BEHAVIORAL TECHNOLOGY LABORATORIES

Department of Psychology

University of Southern California

A multidisciplinary program jointly sponsored by

The Engineering Psychology Programs and The Information Systems Programs
Office of Naval Research

and

The Human Factors Engineering Division
Naval Air Development Center

Under Contract No. N00014-80-C-0493
ONR NR196-165

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Report No. 1	2. GOVT ACCESSION NO. AD-A102	3. RECIPIENT'S CATALOG NUMBER 513
4. TITLE (and Subtitle) DESIGN FOR THE MAINTAINER: PROJECTING MAINTENANCE PERFORMANCE FROM DESIGN CHARACTERISTICS		5. TYPE OF REPORT & PERIOD COVERED Technical Report June 15, — July 1, 1981
		6. PERFORMING ORG. REPORT NUMBER Technical Report No. 95
7. AUTHOR(s) Douglas M. Towne, Michael R. Fehling, Nicholas A. Bond		8. CONTRACT OR GRANT NUMBER(s) N00014-80-C-0493
9. PERFORMING ORGANIZATION NAME AND ADDRESS Behavioral Technology Laboratories University of Southern California 1845 So. Elena Ave., Redondo Beach, CA 90277		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Engineering Psychology Programs Office of Naval Research (Code 455) Arlington, VA 22217		12. REPORT DATE July 1981
		13. NUMBER OF PAGES 90 + iv
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) TR-95, TR-1		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Approved for public release: Distribution unlimited

18. SUPPLEMENTARY NOTES

Also sponsored by Information Systems Programs,
Office of Naval Research and
The Human Factors Engineering Division,
Naval Air Development Center

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Maintainability
Maintenance performance
Troubleshooting
Fault Isolation
Diagnosis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

We hypothesize that the maintenance activity imposed by an equipment may be effectively projected from one or more computed reference maintenance strategies. These include multi-variable strategies (optimum or "expert" approaches), single-variable strategies (time-dominant, reliability-dominant, information-dominant, component-dominant), and a stochastic strategy in which tests are selected at random. The optimum performance strategy and the

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED 411 660

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) random testing strategy provide bounds on the expected maintenance performance. The single-variable approaches are suspected to be reasonable approximations of human activity under various conditions.

The constituents of maintenance performances generated from these strategies, and their associated performance times, are shown to be a direct function of system design. Computed manual times for each of these approaches are presented for one equipment. These preliminary data suggest that maintenance time may be considerably less sensitive to fault diagnosis strategy than expected.

Our work leads us to view a troubleshooter as a strategically flexible, data-driven, and opportunistic problem solver. We describe some recent artificial intelligence models of problem solving which support our conception of the troubleshooter. Such models provide a basis from which the computed strategies described elsewhere could arise.

An interactive, computer-controlled, video system will present maintenance problems to experimental maintainers, to determine if reliable projections of maintenance workload can be made from computed strategies. This configuration allows subjects to direct the maintenance procedure in real time, observe tests being performed, abort tests in progress, and to notice conditions not explicitly sought. These performance conditions are crucial to observation of realistic maintenance performance in an experimental environment.

SUMMARY

Design for the Maintainer:

Projecting Maintenance Performance

from Design Characteristics

We hypothesize that the maintenance activity imposed by an equipment may be effectively projected from one or more computed reference maintenance strategies. These include multi-variable strategies (optimum or "expert" approaches), single-variable strategies (time-dominant, reliability-dominant, information-dominant, component-dominant), and a stochastic strategy in which tests are selected at random. The optimum performance strategy and the random testing strategy provide bounds on the expected maintenance performance. The single-variable approaches are suspected to be reasonable approximations of human activity under various conditions.

The constituents of maintenance performances generated from these strategies, and their associated performance times, are shown to be a direct function of system design. Computed manual times for each of these approaches are presented for one equipment. These preliminary data suggest that maintenance time may be considerably less sensitive to fault diagnosis strategy than expected.

Our work leads us to view a troubleshooter as a strategically flexible, data-driven, and opportunistic problem solver. We describe some recent artificial intelligence models of problem solving which support our conception of the troubleshooter. Such models provide a basis from which the computed strategies described elsewhere could arise.

An interactive, computer-controlled, video system will present maintenance problems to experimental maintainers, to determine if reliable projections of maintenance workload can be made from computed strategies. This configuration allows subjects to direct the maintenance procedure in real time, observe the tests being performed, abort tests in progress, and to notice conditions not explicitly sought. These performance conditions are considered crucial to observation of realistic maintenance performance in an experimental environment.

ACKNOWLEDGEMENTS

The research described in this report was sponsored by Dr. Martin Tolcott and Mr. Gerald S. Malecki, Engineering Psychology Programs, Office of Naval Research, by Mr. Martin Denicoff, Information Systems Programs, Office of Naval Research, and by Dr. P. N. Curran, (CDR, MSC, USN) and Dr. D. K. McBride (LT, MSC, USN), Human Factors Engineering Division, Naval Air Development Center. Their support and encouragement are gratefully acknowledged.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. Introduction

Maintenance activity is a function of three primary factors: the human performer, the environment in which the activity is performed, and the system being restored or adjusted.

The maintainer's capabilities are determined by his innate abilities; his training; the type, recency, and amount of experience; and his motivation. The ease of performing the task can be greatly affected by the environment in which it must be performed. The time constraints, past workload, availability and quality of test equipment, and ambient conditions, such as space, temperature, and visibility, are just a few.

The characteristics of the system itself, however, dictate the inherent difficulty of the maintenance task. The design of the man-machine interface, which may include switches, dials, controls, and test points, determines the ease with which information about the system can be obtained. The internal design (modularity, complexity, accessibility, and so on) determines the ease of identifying and resolving the failure.

This report is concerned with techniques for determining the maintenance requirements imposed by a system's design. Part One explores previous techniques for predicting maintenance workload, cognitive aspects of maintenance, and a summary of some relevant models.

Part Two presents a technique for projecting maintenance performance from a general representation of the system design. This technique yields a set of fault isolation action sequences, each produced according to one of eight general troubleshooting strategies. The approach for computing the associated performance times is described, as is the experimental technique to be employed to determine the functional relationship between observed troubleshooting performance and the general strategies.

PART ONE: PREVIOUS RESEARCH

II. Techniques for Analyzing Maintenance Workload

Background

Maintainability emerged as a true engineering and psychological specialty in the early 1960's. By that time, standard indices of maintainability had been discovered and rediscovered. Most often, these indices were based on the distribution of "downtimes" in the mission cycle. Gradually the idea was accepted that, when a new system idea was proposed, an important part of the proposal would be the estimation of such parameters as mean time to repair. The U.S. Department of Defense was the prime mover in the early work, because of the devastating military experience with new electronics items. Certainly the field maintenance problems are continuing. Here is a recent quote from a leading scientific journal (Smith, 1981).

> The Navy has equipped each of its most advanced ships with a sophisticated radar system that tracks several targets at once and automatically fires the ship's weapons. But it works only 60 percent of the time, because of random failures of its 40,000 parts. The rest of the time, the ships are virtually defenseless.

The Prediction Methods

The techniques which have emerged to date are moderately successful, in general, in producing repair time estimates which correlate with actual repair time data. Unfortunately, the existing techniques tend to be specific to particular technologies or maintenance settings, they tend to offer little insight to the designer, and most tell nothing about the performance required of the maintainer. These maintainability prediction methods can be classified into six categories.

Empirical extrapolation. For a new radar system, one might predict that maintainability requirements will be about as they were on an old radar system that is similar to the new one. Of course, it may be hard to say just how similar the new item is to the previous model, but a rough similarity rule may still be practically useful. At least the real-experience data should introduce some realism into expectations for the new system.

As analysts of maintenance data have noticed, there are a few generalizations that can be made from a casual inspection of time-to-repair data. For one, the mean or median active repair time, for major military electronic systems, is often close to one hour. (This may say why the prediction methods have had any success.) Using more recent field records, Wohl (1980) often found modal times at about the one-hour point, with long "tails" in the repair-time distributions.

A second possible empirical generalization is that variance in repair times among military equipments is largely due to the maintenance concept employed. Airborne radars and radios are serviced via module replacement policy, whereas ship and ground-based items may require troubleshooting and repair down to the piece-part level. Hence, standard deviations for airborne equipment are on the order of half an hour, as compared to about one and a half hours for large ground and ship systems.

A third extrapolation rule is often cited by field users of complex equipment items: the actual time-to-repair _in the field_ is several times higher than the "demonstrated" repair times during system acceptance tests. In fact, a Philco study showed that field times-to-repair were three to four times as long as those observed during demonstrations. These results are corroborated in a recent study by Wohl (1980). Cynics would suggest that neither technicians nor faults involved in demonstrations are representative of field conditions. As discussed later, the experimental environment itself, which includes definition of the fault, typically filters out significant complexities confronted in the field.

A fourth empirical finding is that the distribution of repair times is skewed by a small number of very long times, so that the mode of the distribution is generally far less than the mean. Many early studies have found a very good fit of repair times to a log-normal function (Horne, 1962; Horvath, 1959; Balogh, Hennessy, & Reynolds, 1974). However, recent evidence contradicts this conclusion. Wohl (1980) reports a group of thorough analyses of repair time distributions. Using large samples of field maintenance data reports from Air Force sites, he found that active repair times often were not log-normal, and when plotted on Weibull probability paper, a two-component data process often appeared. For one typical system, nearly 60% of the faults required less than one hour to repair, yet the remaining repair times were so long that the total mean was over three hours.

Critique. As far as we know, procurement offices and contractors do not systematically apply previous maintenance compilations to produce active repair times for a new system. This would require the identification of the several factors that would affect the accuracy of predictions from one system to another. Instead, it appears that many informal projections are made.

Checklist methods. Many factors are known to facilitate preventive and corrective maintenance tasks. Clearly, if some key test points are inaccessible, unlabeled, or otherwise difficult to use, then the equipment will be harder to service. Lists of good design and support features have been assembled, with the idea of scoring a system on the various criteria. The famous Munger-Willis list gave 241 design features which had potential significance for maintainability (Munger & Willis, 1959). A more manageable scheme derives from MIL-HDBK-472 (U.S. Department of Defense, 1966). There are three design check lists in the document. List A is concerned with physical features such as access to and display of information, the types of fault indicators, safety considerations, and so forth. List B treats the need for external facilities (special equipment, etc.). List C evaluates the personnel requirements for successful maintenance, and has items about the demands for logical analysis, alertness, concentration, strength, and manual dexterity. According to some trials at RCA-Camden, reasonable, slightly optimistic, predictions do emerge from the analysis.

Critique. The checklist procedure certainly has one thing to recommend it: the process of scoring the design and support features will bring out serious faults.

Three objections to checklist predictions, however, are (1) the weights, though statistically derived and "objective" for the system originally studied, are seldom cross-validated on other equipments; (2) the design features scored tend to be observable and primarily independent; complicated internal features and interactions tend to be ignored; and (3) the reliability of the predictions made, and of the predictors themselves, is seldom known. For such reasons, it may be well to regard checklist reviews as useful for the internal design staff, rather than as satisfactory quantitative prediction schemes.

Time synthesis simulation methods. Psychologists frequently break down whole tasks into simpler elements. These subtask elements are then separately studied and combined in various ways. If the subtask performance parameters are defined probabilistically, then appropriate distributions of overall performance values can be generated. If the global performance parameters agree well with those observed in the real case, then the model is said to be validated. The synthesis can be further validated, if expected changes in real performance come from experimentally produced changes in the micro-elements.

Several projects have employed time synthesis simulation with generally positive results (Rigney, Cremer, Towne, & Mason, 1966; Siegel & Wolf, 1969; Strieb et al, 1980).

Critique. The concept of time-synthesis simulation is a powerful one. Parameters can be varied easily, and hundreds or thousands of simulated task runs can be quickly computed, so that the (model) effects of possible change can be tried out.

There are challenging technical problems in all parts of time-synthesis simulation. Many problems are encountered in settling the right task descriptive level, in obtaining suitable performance figures from people, and in managing the problems of task correlation and level shifting. Though some complex behavioral routines have a straightforward sequence of subtasks, it is often difficult to synthesize a troubleshooting sequence that resembles human performance. The technique described in Part Two may be considered in terms of the time synthesis technique.

<u>Counting methods</u>. At the extremes, sheer numbers can seem to dominate a maintenance situation. An equipment that has 50,000 parts should be a difficult thing to service. So one indicator of fault-locating difficulty could be the number of hardware elements that the technician has to consider. Information theory can express, for example, the "amount of uncertainty" in a fault-location problem as $U = \log_m N$, where N is the product of defined failure modes times components, and m is the number of possible outcomes of a test.

Of course, much depends on the way that the parts are arranged, and on the possibilities for "block elimination" of whole segments of the equipment. Several projects have tried to combine some notion of the richness of test indications with a parts count. Leuba (1962), for instance, proposed a measure in which maintainability varied directly as the number of elements in the system, and the number of symptoms which can be caused by several different elements.

Critique. Sophisticated counting techniques may yield quantitative relationships between repair time and the counted elements which are useful in projecting the likely maintenance load imposed by a system. It must be realized, however, that the pure counting measures which prove to be correlated with repair time may, in fact, only be indirect indications of system size, scope, and complexity. We might equally expect measures such as system weight or system volume to also provide significant correlations. Thus, most attempts to derive a counting measure incorporate features of system structure beyond sheer number. For example, Wohl's approach cited above seeks to provide a measure reflecting the intuitive notion of the "complexity" of the system structure. We will discuss this "complexity hypothesis" below in more detail.

Cognitive methods. A cognitive approach to projecting maintenance workload postulates specific mental processes involved in troubleshooting and seeks to identify aspects of design which bear on those processes. Such processes might include perceptual or pattern recognition systems, a memory component, as well as processes for inference. Additionally, one may characterize various strategies for troubleshooting in terms of these component cognitive skills, how they are interrelated, and when they are used. Thus, aspects of equipment design may be sought which impact these cognitive strategies via their effect on underlying cognitive processes.

Critique. Complete cognitive models which arise from considerations of the mental processes involved will be exceedingly difficult to develop for practical use in the foreseeable future. Section III explores cognitive aspects of maintenance more fully and considers some of the limitations in cognitive processes which may be significant.

Complexity Measures

It seems quite reasonable to look for some way of describing the "complexity" of the target system and then demonstrating the precise relationship between system complexity and various aspects of maintenance task performance such as Mean Time To Repair, (MTTR). To understand how the construction of a system affects the maintainer, one must have a grasp of how the various aspects of system design are reflected in the task structure for maintenance. Furthermore, to assess the difficulty imposed by this task structure one must have a notion of how it impacts cognition.

Let us first consider the possibility that for each system design there exists some parameter which represents the complexity of the system and/or the task of maintenance on that system. This parameter is normally thought to be expressible as a functional combination of some set of measurable system features (e.g. Rouse, & Rouse, 1979; Wohl, 1980). This parameter could then be used as a predictor of maintenance task difficulty, task completion time, or some other representative measure of maintenance performance. The existence of such a parameter would certainly simplify things. Under the right conditions the designer would be able to take appropriate measurements of a system and provide a useful predictor of the complexity of the task faced by a maintainer of that system. For purposes of the present discussion we will call this set of ideas the "complexity hypothesis".

The research of Wohl (1980) is representative of attempts to use the complexity hypothesis just described. Wohl develops a model which rests upon the notion that troubleshooting involves an enumerative process of searching and testing all components within a suspect set. Since such a search process is dependent upon the complexity of the interrelations among those components, Wohl proposes a measure based upon the product of the average number of component interconnections and average number of electrical junction interconnections. A model is then constructed in which this measure is combined with parameters which estimate the basic diagnostic time factor, and the effect of environment. This model assumes a specific (modified Poisson) distribution of test times at each step. With this model, Wohl is able to achieve a very good fit to data for mean active repair times. Unfortunately, the high correlation (0.97) reported by Wohl between predicted and observed repair times is confounded due to the statistical interdependence of the predicted and observed data points. And since the parameters of Wohl's model, including those supposedly reflecting equipment complexity, are all set to their values by a "best visual fit" of the model to the repair time data, no conclusion can be drawn about the true relationship of complexity to repair time. In fact, values for the complexity parameters of Wohl's model seem to vary significantly less than do values for the other parameters from one case to the next. Therefore, variations in predicted repair times are primarily determined by variations in factors (such as average time to complete an action) more than they are by complexity indices.

Rouse and Rouse (1979) have also looked at the complexity hypothesis, although they have done so from a different perspective than that of Wohl. In their research, the issue of complexity is considered from a somewhat more psychological point of view. The authors review the issue of complexity in terms of the literature on perceptual complexity and problem solving complexity. A number of specific indices are then developed by Rouse and Rouse including one based upon an information theoretic measure of search complexity and another based upon the absolute number of relevant relations among the suspect components. These two indices, in particular, provide reasonably accurate predictions of human fault diagnosis performance.

Nauta and Bragg (1980) have taken a quite different approach to the issue of complexity of design for maintainance. In this study, the authors develop the view that complexity is a multivariate function of system design properties, test attributes, psychological abilities of the repair technician, and the effects of the technician's training and experience. An extensive catalog of variables in each of the above categories is considered, motivating arguments are developed for them, and plausible hypotheses about their effects on the maintenance task are suggested. The approach also avoids any attempt to force the multidimensional issue of complexity into a single variable. However, the approach taken by these investigators is really a descriptive rather than a predictive one, since many of the measures suggested require observation of a fully operational system, a seemingly insurmountable problem for a design engineer interested in a good *a priori* predictor of maintenance workload. Furthermore, the large number of variables considered by these authors are analyzed individually so that one is left with the difficult task of specifying how such measures are to be used for quantitative prediction.

The idea that one might obtain a simple index of system maintenance complexity is an attractive one. It is made plausible by the common sense attitude that there exists some single locus for the difficulty one will have in performing the maintenance task on a given system, and that this effect will be different from one system to the next. However, even if there is indeed a single resultant effect which is reflected in measures such as repair times, it does not follow that the causal locus of this effect may be found in some unitary aspect of the physical system. Rather, it is possible that a number of independent factors in a design contribute to the task difficulty, which is consolidated into a single effect only as a result of the action of specific psychological processes.

III. Cognitive Aspects of Maintenance

The notion that there exist performance limitations for the component cognitive processes implies the potential for error, or, at the least, inefficiencies in the conduct of a maintainer using these cognitive skills. These cognitive skill limitations and the potential for error force us to reconsider what it is that constitutes rational performance. For example, the performance of some apparently redundant diagnostic test might make sense as a means of cross checking previous results. The maintainer may or may not be aware of this. In any case, the potential for error alters the demands of any problem solving task such as may be presented to a maintainer.

Cognitive Processes

Pattern recognition. Pattern recognition processes are clearly important to performance of many components of the maintenance task. For example, the maintainer is required to interpret specific perceptual data as indicative of correct system operation or system failure, to recognize pattern data during performance of diagnostic tests, and to recognize the complex patterns which determine various states of system and subsystem configuration during visual inspection procedures. Moreover, it is well established that important aspects of the diagnostic procedures involved in troubleshooting can be characterized as a type of pattern recognition process. The well known research of deGroot (1966) has looked at human chess experts engaged in a paradigm example of "high level" problem solving activity which apparently depends upon processes such as strategy formation and search among a number of alternative steps toward problem solution. However, deGroot concludes that much of what appears to be "higher level" problem solving activity is largely a function of processes which respond to the identification of complex patterns in the problem data. Pau (1981) specifically demonstrates how various techniques from the pattern recognition literature can be applied to aspects of troubleshooting. Giascu (1977) presents a model which derives an optimal strategy for diagnosis using techniques taken from information theory.

The human factors research reveals a number of ways in which the maintainer could be potentially affected by specific design features which impact perceptual and pattern recognition processes. Much research has been done recently to investigate the conditions which affect the human operator engaged in the task of system monitoring (cf., Rasmussen & Rouse, 1981). Perhaps one moral of the story told by this research is that the maintainer benefits most from receiving neither too little nor too much data. If system monitoring must be continued for long periods of time, to enhance the probability of fault detection, then it is well established from signal detection research that factors such as attentional loss will detract from performance. On the other hand, it is equally problematic for the operator to receive an overabundance of signals on system performance, especially if action must be taken in response to these signals (Boecek & Veitengruber, 1976; Cooper, 1977). Note that this point about a potential surplus of input is relevant to the conditions which prevail during the diagnostic phase of maintenance, as well as during the monitoring phase. For example, perhaps fault diagnosis performance will be more efficient if symptoms are predominantly normal or predominantly abnormal. A number of other perceptual factors are known to be important. For example, the difficulty human perceivers have in detecting rare or otherwise unexpected events may perhaps be due to the use of analysis by synthesis (Neisser, 1967) processes in pattern recognition according to which the identity of perceptual input is determined from a relatively small initial sample of its input features which are then used to "synthesize" (i.e., infer) the nature of the remaining larger proportion of its identifying features. Further examples include the discriminability of fault signals from the background (correct state information) and the stability or instability of patterns over time which are indicators of system states. One very important point to make is in regard to the effect of redundancy. The perceptual literature and the recent research on reading both indicate that an optimal amount of perceptual redundancy is an aid to the efficiency of the perceptual process (cf., Haber, 1978). Finally, we note that the dominant view of human perceptual process has for some time proposed that pattern recognition processes are based upon a complex feature detection system (DeValois & DeValois, 1980).

Attention. An enormous amount of research has been devoted to the topic of attention. For the most part, the prevailing view of attentional processes begins with the concept of selective attention (Moray, 1970; Norman, 1976). The main theoretical question of interest has been to determine the nature and locus of the selector process. A recent and popular view is the thesis that negative attentional effects are a result of task demands exceeding the processing resources available to the subject (Norman & Bobrow, 1975). These processing resources include resource driven (top down) processes which direct attention on the basis of knowledge the subject already has about the current task conditions, and data driven (bottom up) processes which are responsive to new information gained from the array of input. Moray (1981) has provided one of the few discussions of attention addressed to the topic of maintenance activities. He notes that, unfortunately, most of the research on attention is of little use for one interested in its relationship to subjects working with large scale, real world systems due to the complex, (continuous) multivariate, non-linear, and highly structured nature of these systems compared to laboratory paradigms. Nevertheless, Moray is able to draw some tentative conclusions about attention in the maintenance task from the literature and his own research. For example, he mentions the number and complexity of sources (e.g., displays) from which the operator must obtain facts as critical to operator performance. Moray also discusses the importance of "predictor displays" by which he means the display by the system of variables which will indicate that some system component will remain in a certain state for a significant period of time, thus allowing attention to be shifted temporarily to another system component.

Moray also points out the importance of attention for diagnostic processes. For example, he notes that "<u>following an abnormal observation, highly correlated sources should be sampled</u>". By this he refers to the fact that in a complex equipment the observation of evidence that the system is in a failed state will usually come from some subsystem such that the maintainer should probably restrict subsequent observations to that subsystem. This of course, may not always lead directly to the fault, since the evidence of failure might be displayed in a module which is performing correctly but is responding to erroneous input received from a different subsystem. An obvious example is the detection of an erroneous display of data on a computer system's video display terminal. This evidence of failure could, in fact, be a result of a defect in some distinct subsystem such as a disk storage system, the failure of which could be propagated to the contents of the terminal. It might be that Moray's dictum is not applicable in many practical cases such as this. In fact, the challenge to the maintainer could well be the reverse--to not let attentional resources become too focused at the wrong point in a diagnostic procedure. In any case, it is clear that attentional factors can clearly affect the performance of the maintenance task.

Memory. Despite a huge amount of research devoted to memory, the psychological community is still somewhat divided on the basic question of how many distinguishable memory stores exist. Extant research establishes a short-term memory or span of attention effect which is distinguishable from effects due to use of general and more permanent types of knowledge, regardless of whether this effect is best explained in process or storage terms. Thus, in what follows we will allow the distinction between short-term or "working" (post-perceptual) memory phenomena and phenomena involving relatively permanent knowledge representation structures and processes. For a task such as that performed by a maintainer, both working memory and permanent memory processes may be impacted. For example, fact retrieval abilities are required representing knowledge of the possible actions which may be taken, the procedural knowledge of these actions, constraints on these actions imposed by the current situation (including the history of actions already taken and the results obtained), the knowledge of the target system's structure and function, and recall of the facts thus far obtained during the task. From the research literature one can expect that these processes of fact retrieval will be affected by such things as the extent of hierarchical organization of knowledge about the task and the system, and how well the various events which can occur will provide good cues to retrieval of all and only the information needed in a particular context. Second, the capacity limitations of working memory have long been well established (Miller, 1956). The existence of such limitations restrict the maintainer's ability to keep track of all the needed facts at one time. For example, if an extensive amount of procedural information were needed to perform some subtask, other facts, such as where the maintainer is in a previously constructed plan, could be lost. As is true for attention, the memorial resources are limited, and this predicts difficulty for a maintenance task in which procedures to be performed are unduly complex. Similarly, the necessity to process extraneous data or engage in necessary, but irrelevant, tasks (e.g. moving things out of the way) while performing some action could, for example, divert a maintainer from one course of action to another which is less reasonable or even innappropriate in light of the current context. Another useful point is to distinguish between recall of knowledge held prior to initiation of the current maintenance task and knowledge obtained during performance of the task. We have observed that technicians often have difficulty recalling exactly what symptoms have been observed, and under what conditions, as a task proceeds. This problem occasionally is so severe that the testing must be virtually reinitiated. Thus, recalling what is known about system structure and procedural sequences may be less critical for continued task performance than recalling what has and has not been done. If this is true, then short term memory factors might play a most critical role in maintenance task performance.

Inference. It is clear that inferential processes such as inductive and deductive reasoning are essential to many performance aspects of maintenance. This is particularly true during troubleshooting. The maintainer must be able to reason deductively, when required, to determine whether a set of test outcomes is sufficient to uniquely isolate a failure. Inductive reasoning is required, for example, when the maintainer must select the next test to perform, based upon available data regarding individual probabilities of component failure in combination with what is known so far about symptoms.

The research on human inference and reasoning processes, and on problem solving is quite typically restricted to laboratory paradigms which bear little, if any, direct relation to the concrete reasoning required of a maintainer. However, there are some general patterns of results which are quite useful in consideration of the maintenance task. First, the excellent analysis by Amarel (1968) demonstrates the importance of how a problem is represented. This is related to the older result of "functional fixedness" (Duncker, 1945; Maier, 1931) in which the problem solver has difficulty achieving a solution to a problem because he is unable to see that some object may be used in a novel way to help construct a solution. Representational effects are a possible source of maintenance task difficulty in terms of the extent to which a system design hides important clues to the nature of a system failure. As an example, a failure in a component may be quite difficult to isolate if the manifestation of that failure appears most prominently in a physically distinct component.

Given some form of problem representation, what reasoning processes are available to the problem solver? First, there is an ongoing debate among those who claim evidence that humans reason illogically, citing the frequency with which human reasoners embrace invalid argument forms (e.g., Chapman & Chapman, 1959; Pezolli & Frase, 1968), and those who claim that humans are, in fact, quite logical in their argument structures but err in the way they encode the information to be used in these structures (e.g., Henle, 1962; Revlis, 1975a, 1975b; Mayer & Revlin, 1978). Unfortunately, the fundamental issue is not yet decided.

Complementing this research is the work on diagnostic judgement. This latter research investigates the diagnostic conclusions of both novices and experts as a function of the features of the evidence provided from which to make the diagnosis. The most notable results here are to be found in the work of Kahneman and Tversky (Tversky & Kahneman, 1974; Kahneman & Tversky, 1979). They have observed some general principles to which human decision makers tend to adhere. The first of these is the "representativeness heuristic". According to this principle, the question, "will event A be generated by process B?", will be decided affirmatively to the extent that the event A resembles process B. According to this principle, if failure in a computer disk drive is manifest at the video display terminal, the troubleshooter is more likely to generate hypotheses of failure in the display than in the disk storage system. Another principle proposed by Kahneman and Tversky holds that there is an "anchoring effect" in that evidence obtained early by a diagnostician will create a starting point from which subsequent evidence will move the diagnostician only with great difficulty. In terms of the maintenance task, this principle claims that initial front panel evidence that is misleading (perhaps due to the representativeness heuristic) will have negative consequences for the likelihood of rapid fault isolation. Finally, Mynatt, Doherty, and Tweney (1977) produce evidence that human troubleshooters perform according to a bias to confirm a suspicion rather than test and eliminate hypotheses. All of this leads to a rather interesting question: Is it advantageous in any way for human problem solvers and reasoners to use these hit and miss methods of reasoning? It is possible that the logically deficient methods are in fact quite productive in the context of real world task demands. In the case of maintenance it would be interesting to analyze the efficiency of such strategies from this perspective.

Planning. The planning processes in maintenance are not well understood. Unlike problems in which all the required information is presented at the onset, troubleshooting proceeds from a small fraction of the data required tc ultimately isolate the fault. Generally, little benefit is derived from formulating extensive contingency plans prior to performing a test, since much of that planning concerns results not subsequently obtained. Thus troubleshooting may be a process in which actions are selected based upon factors such as ease of performance, rather than upon the extent to which they meet other technical criteria. At a higher level, the maintainer somehow allocates his time, weighs competing influences, and decides when to shift to a more promising attack on the problem.

Until recently, there was very little research directly investigating how it is that people construct and use plans to guide their activities. The field of research which has given the most explicit attention to planning is perhaps Artificial Intelligence (A-I). This is, no doubt, a result of the need to devise powerful control processes for the various machine systems intended to perform complex tasks. Unfortunately, this A-I research on planning processes was unmatched by similar efforts in psychology to determine the nature of human planning and control processes. Hayes-Roth (1980; Hayes-Roth & Thorndyke, 1980) has recently produced some excellent data (as well as a model to be discussed below) on human planning and control processes. The results of this work demonstrate a number of interesting features of human planning processes. First, the human planner appears to be data driven and opportunistic. Interim plans are formed and altered during conduct of the task on the basis of new information obtained during execution of earlier versions of the plan. Planning seems to involve problem solving at a number of levels of abstraction, and the planner will make decisions regarding planning at all these levels. Finally, the planner will typically underestimate the amount of time which will be required to carry out the various components of a planned task. Hayes-Roth contrasts this view of planning with that developed in a typical A-I model.

Human Error

A further implication of the existence of such cognitive constraints as are being discussed here has to do with the potential for error. It seems clear that humans are prone to error in performance of even basic skills. (It is currently popular among cognitive scientists to view even underlying cognitive processes as skills.) There is good reason to believe that maintainers, even expert ones, are quite error prone. Recently, Norman (1979, 1980) has investigated errors in human performance of a variety of tasks. Much of Norman's data is anecdotal or based on uncontrolled field observations; however, this research strongly demonstrates the clear tendency to err during performance of even well practiced tasks. Norman distinguishes between two types of error in performance: errors in the formation of an intention are termed "mistakes"; errors in the execution of an intention are "slips". Norman is careful to point out the fact that mistakes and slips can occur "even when the person has full information of the state of the situation".

Norman's distinction between mistakes and slips points to the potential of error at all phases of a task, including the problem solving and planning activities which underlie action. Third, the ubiquity of error suggests that human problem solvers will operate with some knowledge of its possibility. For a task such as maintenance, this implies that certain actions which appear to be non-productive may in fact be quite productive.

IV. Models of Troubleshooting

We turn now to a discussion of some representative models of troubleshooting, in order to reveal assumptions typically made about the nature of the task of troubleshooting, the cognitive skills which underlie the task, and the way these skills are used. Let us first briefly mention normative models of troubleshooting.

Normative Models

In a typical normative model, the sequences of diagnostic tests and component replacements are constructed by use of a choice function which calculates each step in a sequence from some measure of the relative values of the choices available. The choice function may be based upon such factors as information yielded by the alternative tests, costs of performing tests, and reliabilities of system elements. In a model based on a pattern recognition criterion (Giascu, 1977), the next step in a diagnostic procedure (test or replacement) is determined by calculating which next step will return the most information given a particular set of procedures and results produced up to that step. This function can be calculated from the reliabilities of the components, and the relationship between tests and malfunctions such as may be obtained from a symptom-malfunction matrix. Similarly, the BETS model (Rigney, Cremer, & Towne, 1966) chooses the next best test by using a choice function derived from Bayes theorem in probability theory applied to the component reliabilities.

Optimal strategies produced by normative models serve to specify potential lower bounds for maintenance time. Such models provide idealized task sequences to be performed by a maintainer. While such models say nothing directly about conditions which invariably lead to departure from optimality, they can be regarded as a baseline by which designs might be compared. A set of eight such normative models, including the optimum strategy, form the basis of a technique described in Part Two.

A-I Models

The proportion of A-I research directly investigating troubleshooting is very small compared to the vast amount of research done on other forms of problem solving. So it is no surprise that those efforts to model troubleshooting have been applications of the techniques used to model other forms of problem solving.

The following basic conception of a problem solving system typifies the view held in an A-I model. The problem solver has some representation of the current state of affairs (which we call the INITIAL STATE) and also a representation of some desired situation (called the GOAL STATE). The problem solver also is capable of enacting any members of a set of procedures (we will call these TRANSFORMING PROCEDURES) which may be used to transform one state of affairs into another. The problem solver is then required to specify and enact a sequence of these procedures (we will call this a SOLUTION SEQUENCE) such that the initial state is changed into the goal state. This characterization is a reasonable, though very general, approximation to most problem solving systems in A-I. From it we see that the performance of the problem solving system is determined by the way states of affairs are represented, the nature of the possible procedures for transforming states of affairs, and the techniques available for selecting a solution sequence from the set of transforms. The bulk of A-I research on problem solving has focused upon methods for finding solution sequences. It is this work which we now discuss, following the general outline of Nilsson (1979). In some of the early problem solving systems (Raphael, 1971) the transforming procedures were enacted upon occurrence of a specific condition in the current state of affairs. So, in such a system the initial state will contain some condition which causes some procedure to be enacted. This procedure will produce changes, creating a new state of affairs which differs in some details from the previous one. One can expect a goal state to be reached in such a system only because the transforming procedures are designed so that, on the average, a state of affairs following enactment of a procedure will be "closer" to the goal state than that upon which the procedure operated. This type of problem solver is relatively flexible in that an appropriate procedure can be enacted any time the right conditions prevail. However, the system can get into serious problems, such as infinite loops, if the condition and procedure relationships are not properly specified. This is an intentionally oversimplified description, and actual systems using this technique have added special features to help alleviate some of the inherent difficulties. Nevertheless, it is clear that this approach places all the "intelligence" in the prearranged condition-action relations given by specification of the transforming procedures. The selection of a solution sequence in such a system is a fortuitous result of clever arrangement of these condition-action relations.

A more sophisticated approach to the selection of a solution sequence is found in those problem solving systems which, a) provide for planning whole sequences of action before their execution, and b) engage in a kind of backward reasoning. The idea behind planning is that the problem solver has stored a representation of the effects of each potential action on any state of affairs. Having this capability the system can "simulate" enactment of a procedure and store a copy of the state of affairs which would result from its enactment. Using this ability to simulate the results of actions, the backward reasoning procedure of goal reduction operates as follows. The goal condition is noted, and the system selects a procedure which, if enacted, would result in the existence of the goal condition. This procedure will be enacted, of course, only if certain preceding conditions exist. These presupposed conditions are then each listed as goals to be achieved and the whole process is repeated. This backward process continues until the preconditions noted in the stored plan structure contain the conditions which exist in the initial state. By then executing the plan structure forward from the initial conditions to the goal condition, a solution sequence of procedures is enacted. Since a condition might be realized by enactment of more than one procedure, a tree structure of possible paths to a solution sequence may be created. If the problem is complex, this planning tree will become huge and the problem solving process will not operate fast enough for any practical application. The solution usually attempted for this "combinatorial explosion" has been to search the tree in a depth-first manner, using some heuristic choice function to select the most productive branch to follow at each point, but this is a partial solution of the combinatorial problem. Nevertheless, the use of goal reduction and planning is a distinct improvement over the previously discussed approach. Approaches like this notion of goal reduction have been used in some important problem solving models (cf., Newell & Simon, 1963, 1972; Fikes & Nilsson, 1971).

A recent A-I development employs "successive refinement" to reduce the need to search a very large space of possible solution sequences. This system is able to represent and make plans about higher level specifications of procedure sequences by temporarily omitting the specification of actions to achieve a selected subgoal. The system simply assumes that such a sequence can be found later. This amounts to producing a plan for achieving the goal which can be filled out later by going back and determining the precise nature of the assumed action sequences (perhaps in various parts of the plan). This technique can, of course, be repeated for any number of hierarchical levels, and by its use the problem of having to search among an enormous number of possibilities can be somewhat alleviated. If the assumption that a condition can be later supported by an action sequence proves wrong, then the system is forced to fail or else to employ some special technique such as backtracking in order to recover.

Some interesting A-I models have been developed, using the techniques sketched above, which are specifically applied to tasks performed by a maintainer. Brown (1977) has proposed a model, called Watson, which troubleshoots defective radio circuits. Watson takes as input a plan structure for tracing a target system's functions which actually represents the functional design of the target system. It resembles a plan structure created by successive refinement techniques in that it is a hierarchical, nested structure of levels of design description (which Brown calls "plan fragments"). This design description is input to a recursive fault localization process which starts at the top level of a plan and isolates a fault to one of its substructures. This process is recursively repeated on substructures until the fault is isolated to a circuit component and the hypothesis of that component's failure is verified. If, at any time, Watson finds that its hypothesis producing mechanism has isolated to a non failed part, it must backtrack to a higher planning level and try another path. The rules Watson uses to isolate a fault include one which uses facts about the qualitative, cause and effect relationships between components to trace backward along paths representing the propagation of effects in the target system's operation. The system representation is tightly hierarchical and so Watson's processing is organized in a top-down manner from the upper, abstract levels of system design through successive refinements of system specification to the lowest level of individual components.

Perhaps the most well known example of this sort of problem solving system is NOAH, (Sacerdoti, 1975) which controls its activity by the use of a planning system called procedural nets. NOAH is meant to be capable of tasks such as disassembling and assembling an air compressor, although its methods are considered to be applicable to a wide range of problem situations including natural language understanding. NOAH formulates its problems in terms of high level goals and decomposes these goals into lower level ones. Each goal specifies sequences of actions. When the problem reduction is complete NOAH has produced a correct plan represented as a partial ordering of elementary actions. The discussion of the successive refinemment technique given above was really a simplified version of how a system like NOAH operates (cf., Nilsson, 1979). NOAH is an excellent plan generator and performs impressively on the demonstration problems it is given. However, such a system makes some very serious assumptions about the nature of an effective problem solving system. For example, a system such as this generates more or less complete plans before execution, thus requiring availability of sufficient knowledge of the target system to simulate its essential functions. NOAH has no explicit knowledge of its plan generating contingencies, therefore, the problem context will affect the planning process in a manner that is predetermined by the implicit features of the planning rules. Such a system cannot, for example, tie together two plans with parallel contingencies. Although it seems easy to add, there is no provision in NOAH for backtracking. Finally, NOAH can easily miss the possibility of an interaction between two separate actions at some depth in a plan, since the representation for these actions are not compared interpretively or in terms of effect.

Goldstein (1974) developed a system called MYCROFT which automatically debugs a class of programs in the high level programming language LOGO. Goldstein's system accomplishes its task essentially by comparing output of a program with a model of its intended effect, and uses knowledge about potential problems in linking together certain program components. Brown claims that this system is inferior to Watson because it can functionally represent its target system only by "running" (simulating completely) that system; however, the comparison of models to debug computer programs and models to repair faulty physical systems is one that deserves further consideration (cf., Wescourt & Hemphill, 1978). A more recent project by deKleer (1979) has extended and refined research such as Brown's. DeKleer's system can construct a mechanism graph for the functional topology of a circuit from a description of its physical topology and identification of its components. It accomplishes its task by analyzing the qualitative nature of local relations among components. The system successively refines a functional representation and chooses among candidate interpretations by selecting that interpretation which determines a purpose (similar to Watson) for all components. DeKleer uses the term "envisionment" to refer to the process of qualitatively simulating the high level functional relations among elements of a system.

Psychological Assessment of the A-I Models

Although neither Watson, nor NOAH, nor any of the other work mentioned is explicitly intended to be a psychological model of the relevant problem solving processes, such work provides a source of ideas for proposing, say, a model of the human troubleshooter. However, from our earlier review of psychological issues certain basic assumptions of these models are suspect. First, we have noted that even experienced human troubleshooters frequently lack extensive and detailed understanding of the functioning of the target systems. Thus, to the extent that the above models invariably require a detailed functional representation of the target system, they are inaccurate as models of the human troubleshooter. At least the provision should be available for the model to upgrade its representation as it gains exposure to the target system. Second, the method of planning by successive refinement, although a powerful one for these applications, is just not what people seem to use. We have discussed the results of Hayes-Roth (1980) which reveal human planners to be far more opportunistic and flexible in their approach to planning than the above systems. And perhaps this is also a much more powerful approach in general.

Finally, these models do not consider their internal computation time in weighing alternative solution sequences. Human performers seem to consider, or at least avoid, effort and time expenditure in both selecting and performing actions. For example, few technicians would invest more than a few minutes in deciding which of two tests to perform if the two alternatives could be performed in just a few seconds.

One approach to problem solving which is responsive to some of these criticisms is the Hearsay-II speech understanding system (Erman et al., 1980). Hearsay has been used in a successful model of human planning (Hayes-Roth & Hayes-Roth, 1979). In a Hearsay model, the problem solving system is composed of numerous, independent "specialists", each of which is a procedure created to do some quite specific part of the overall task. Though these processes are independent, they may interact with each other via a device called a "blackboard" by writing information to that blackboard which may be used by other specialists. More precisely, a specialist is enacted as a response to the presence of specific information on the blackboard. If an appropriate pattern is present on the blackboard, a specialist will be enabled and may then be executed in response to a set of control procedures which handle prioritizing and scheduling of execution for enabled specialists. When executed, a specialist may alter some patterns of information on the blackboard in addition to performing other types of operations. This newly created data on the blackboard will enable the operation of new specialists, which may further alter the blackboard contents, thus enabling further specialists, and so on. In addition to those specialists devoted to operating on problem domain data, other specialists may be devoted to performance of control tasks such as scheduling when other specialists will be enacted. In this way, the control structure, the representation structure, as well as procedures which apply to the problem domain are all equally visible to the problem solving process. In a Hearsay model procedures at all task levels from the lowest level to the most abstract can be allowed to influence each other if this is appropriate to efficient task completion. A model of this type can perform a task opportunistically in the sense that it need not devise a complete plan before acting, and it may alter its course of action radically in response to new data. It appears then that a Hearsay type of control structure could be used as a basis of a model of a troubleshooter with the psychologically appropriate characteristics of flexibility, opportunism, and lack of dependence on an elaborate initial knowledge of the target system. Such a model would provide a rich source of hypotheses about such issues as the effect of system design characteristics on the maintainer's performance.

PART TWO: CURRENT RESEARCH

V. An Analytic Approach to Projecting Maintenance Workload

In developing a technique for assessing the impact of a design upon the maintainer, we are primarily concerned with projecting what work must be performed to meet the maintenance requirements. Subsequent analyses of these characterizations, to evaluate performance time or difficulty, for example, are manageable problems once the consituents (of the performance) are specified.

An ideal technique would project maintenance performance across a wide range of proficiency and environmental levels, allowing designers and planners to evaluate the sensitivity of the design to those variations. Such a technique would reflect the variations in maintenance efficiency as well as the possibly more significant variations in error commission, error severity, and error detection.

A more attainable approach, pursued here, compares and evaluates designs based on projections of error-free performance in a nominal environment. Such a capability may provide the basis for extrapolating to fallible performance at a later time. The need to ultimately confront human error is clear. Maintenance performance is affected not only by the actual commission of errors, but also by the possibility of their commission. Furthermore, alternate designs may present quite different error exposures, which would go unrecognized by an analysis which excludes error.

The variations of possible error-free performance are, of course, immense. At one extreme is optimal performance; the strategy employed minimizes the time expected to find and resolve a failure. At the other extreme is a strategy in which tests are selected at random; no consideration of efficiency is made. In the field a vast array of non-optimal maintenance task sequences are performed which reflect variations in individual skills, training, and abilities. We have formulated eight generic troubleshooting strategies in this domain, which, when applied to a specific representation of a system design, generate troubleshooting action sequences. Times to perform these sequences are then computed by retrieving and accumulating predetermined, standardized motion times for the actions involved. Each performance sequence and time, therefore, reflects the total impact of the system design upon the *maintainer*, if he were to follow the particular strategy.

Subsequent experimentation will be conducted to establish the relationship between observed maintenance performance and the performance generated by the generic approaches. We hypothesize a reliable relationship between one or more of the generic approaches and observed performance.

System Representation

To represent a system design, we require (1) a characterization of the symptom information regarding the state of the system, which can be accessed by the technician, (2) data expressing the "cost" of acquiring that information, (3) reliability data, and (4) a representation of the physical structure of the system.

The first three of these can be organized as a matrix as shown in Figure 1. The columns in the body of the matrix represent replaceable units (RU's) while rows represent tests. Each cell

	REPLACEABLE UNIT					
TEST	1	2	3	4	5	6
1	S11	S12	S13	...		S16
2	S21					
3	S31					
4	.					
5	.					
6	.					
7						
8	S81					S86

TIME
T1
T2
T3
T4
T5
T6
T7
T8

R1	R2	R3	R4	R5	R6

Figure 1. Symptom-malfunction Matrix with Test Costs and Unit Reliabilities

entry, S_{ij} , expresses the consequence upon test i of a failure in RU_j . An entry of zero indicates no effect, i.e., test i is unaffected by RU_j . A non-zero entry indicates an abnormal symptom. Costs of performing each test are entered in the column tab at the right of the array and RU reliabilities are entered into the lower row tab. We will use time as the measure of test cost, but we recognize that the maintainer may continually weigh time cost, dollar cost, personal effort, personal safety and other factors in selecting tests.

The physical structure of the system will be represented as an assembly specification as shown in Figure 2. All system elements appearing in the first (leftmost) column are accessible to the maintainer; the time to remove and replace each is entered in the last column. An element appearing in the second column is accessible only by first removing the element which appears above it in the first column, and so on. Tests are included in this structural representation to indicate what disassembly must be accomplished to initiate each. The test times shown in Figure 1 are therefore the inherent times which are independent of preceding work.

Task Representation

To effectively relate design characteristics to their impact on maintenance activity, a generic structure of activity elements was formulated (Figure 3). The elements in this decomposition relate rather directly to identifiable design issues.

Status identification (SI). All activity performed to determine what fault, if any, exists in a system or equipment is

ASSEMBLY LEVEL				
1	2	3	4	TIME (MIN.)
MODULE 1				.45
	4 COVER SCREWS			2.12
		CKT BD A		.23
		CKT BD B		.36
			TEST 4	.36
			TEST 7	.19
MODULE 2				.38
	TEST 2			.36
	CKT BD A			.36
		Q3		12.44
		R5		9.35

Figure 2. Assembly Specifications

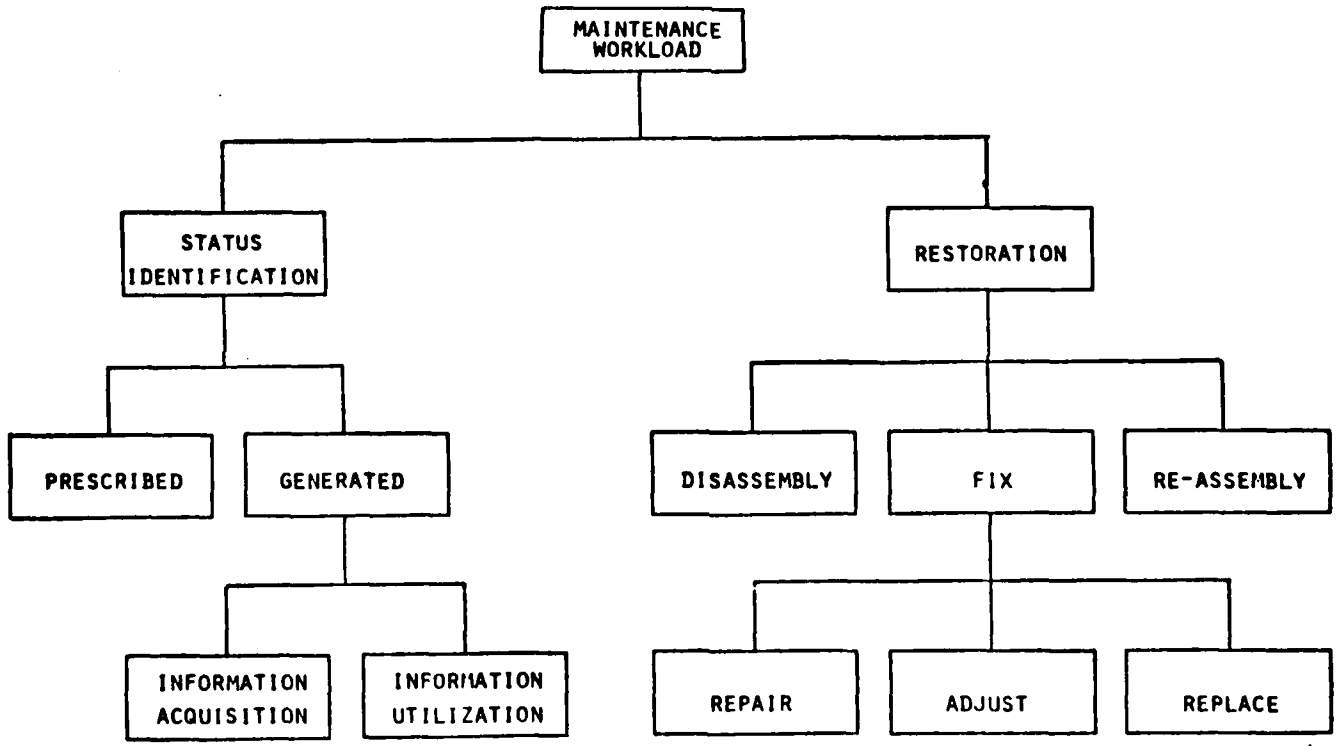


FIGURE 3. COMPONENTS OF ACTIVE MAINTENANCE WORKLOAD

termed "status identification" (SI). This activity is further decomposed into "prescribed activities" and "generated activities". Typically, troubleshooting begins with the performance of a fault verification/localization procedure which is prescribed in the technical documentation. This may involve executing BIT routines, performing manual front panel tests, and/or other well defined procedures.

Sometimes the prescribed SI activity terminates with a successful identification of the fault. In other cases, either the prescribed process fails to identify a single possible cause, or upon making the indicated repair or replacement, the technician finds that the problem persists.

Generated activity commences when the technician begins deciding what action will be taken to locate the fault and/or verify system operation. This is decomposed into two major types, "information acquisition" and "information utilization".

<u>Information acquisition</u> elements are all of the generated activities performed to obtain information about the status of the equipment. These include visual inspections, performing front panel tests, attaching and using peripheral test equipment, and altering the system configuration in order to test or exclude various functions. These manual/perceptual elements are directly affected by the design of the man-machine interface, the ease of reconfiguring the system, and the design of peripheral test equipment.

<u>Information utilization</u> elements include all the cognitive activities associated with generated SI. These include evaluating

symptoms, studying technical documentation, deciding what test to perform next, as well as planning and managing resources. These elements are directly affected by such factors as the relationships between the available test and the internal structure, and the quality of the technical documentation.

Typically, generated SI consists of a sequence of alternations between information acquisition and information utilization. The transitions are not necessarily instantaneous or complete. For our purposes, however, we will regard information acquisition as entirely manual/perceptual, except for the simultaneous cognitive attention required to direct the work; and information utilization as entirely cognitive, except for the incidental manual/perceptual work associated with studying technical documentation.

Restoration. All activity performed to correct the actual fault is termed "restoration". This may include (1) disassembly, (2) replacement, adjustment, or repair of the faulty element, and (3) reassembly.

Two types of activity resist simple classification: (1) any disassembly/reassembly, which occurs as part of SI, and (2) any replacements and adjustments made as part of SI. Some disassembly and reassembly is often performed during SI. This may be done to gain access to additional test points, to facilitate visual inspection, or to accomplish replacements or adjustments performed to isolate the fault (e.g. swapping a cable or trying an adjustment). To assign all such effort to either SI or restoration could seriously distort the analysis of a design. A reasonable procedure is to assign to SI all disassembly, adjustment, and reassembly effort not required to correct the true fault. Thus SI time reflects all activities performed to identify the fault, and restoration time is unaffected by the manner in which the fault is identified.

The foregoing characterization of maintenance activity places no constraints on the order in which various activity types may occur, nor on the number of different occasions, within a problem, that any particular type may be performed. Table 1 indicates some possible combinations of maintenance activity types.

Predicting and Quantifying Performance

Fixed sequences. The actions required to accomplish Prescribed Status Identification and Restoration are predictable from the technical documentation plus the specification of Figure 2. While individual technicians may differ in work pace and efficiency of performing, the technical documentation and system design constrain the actions which can correctly be performed.

The time data for performing tests and assembly/disassembly actions may be based upon estimates, micromotion analysis, or a mixture of these. Estimates would be used when design specifications are not detailed, or when highly precise results are not required or justified.

Micromotion analysis is the synthesis of a defined task from small, pre-analyzed motions (Karger and Bayha, 1966). While this approach yields accurate results and detailed motion documentation, it requires considerable training and application effort. An example of a micromotion analysis of connecting a coax connector to

RESOLUTION OF MAINTENANCE REQUIREMENT	- - - - - - - M A I N T E N A N C E A C T I V I T I E S - - - - - - - - -					
	PRESCRIBED S.I.	INFORMATION ACQUISITION	INFORMATION UTILIZATION	DISASSEMBLY	REPAIR/ ADJUST, REPLACE	ASSEMBLY
Prescribed Fault Isolation Sequence Failed; Technician Located and Fixed	X	X	X	X	X	X
No Prescribed Fault Isolation Procedure		X	X	X	X	X
Prescribed Fault Isolation Sequence Located Fault	X			X	X	X
Fault Symptoms Were Obvious; Restoration Began Immediately				X	X	X
Prescribed Fault Isolation Failed, But Technician Recognized Fault	X		X	X	X	X
False Alarm - BIT Verified System was Operational	X					
False Alarm - Technician Verified System was Operational		X				
Obvious Fault - No Disassembly Required					X	
BIT Indicated Adjustment Necessary	X				X	
Front Panel Checks Indicated Adjustment Necessary		X			X	

Table 1. Typical Combinations of Maintenance Activity Types

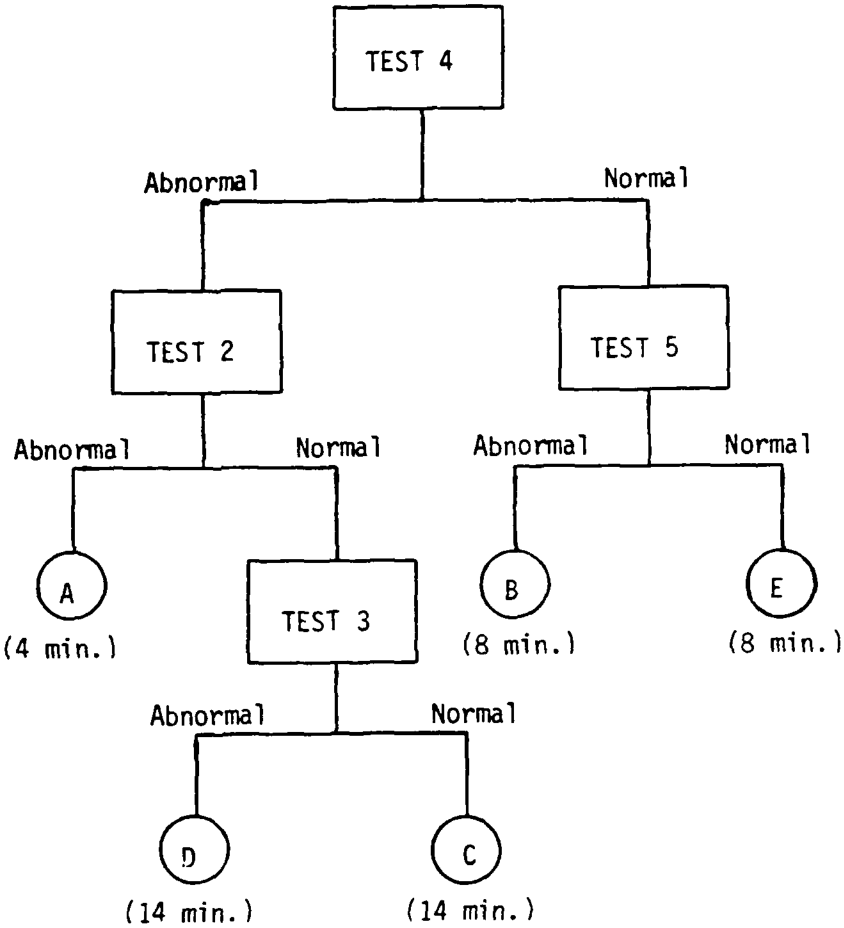
a receptacle is shown in Figure 4. Fortunately, a wide variety of testing, assembly/disassembly, and repair operations have been analyzed and documented in task time data banks. Consequently, a time value for a task may be retrieved from such a catalog, rather than being built up from detailed motion analysis. An automated technique, similar to one now used in industry (Towne, 1968, 1980), will be developed as part of this research to further facilitate this data retrieval process.

Variable sequences. A family of eight primitive troubleshooting strategies has been formulated to represent variations in troubleshooting approach. When applied to a representation of a system, these strategies produce fault trees whose structure and performance time cost are a direct result of the system design (as well as the underlying strategy which produced them).

For each strategy, the selection rule is applied to select the first test. The symptom-malfunction matrix then indicates what system failures would give a normal indication and which would cause an abnormal indication for that test. The selection rule is again applied to each resulting subset, and so on, until a complete fault tree is developed (Figure 5). The time cost of isolating each element is then computed as the sum of the times of all tests which appear in the branch terminating at the element. The measure of effectiveness of a fault tree is Expected isolation time, computed as

DESCRIPTION	MOTION SYMBOL	TIME (MIN x 1000)
1. Reach to Coax Connector	R14B	8.6
2. Grasp Connector	G1A	1.2
3. Move Connector to Receptacle	M14C	10.1
4. Move Connector onto Receptacle (edge hits pin)	P2SSE	11.8
5. Turn Connector to Engage Pin in Slot	P2S3	9.7
6. Release Connector	RL1	1.2
	TOTAL:	42.6 (.0426 minutes)

Figure 4. Micromotion Analysis - Attach Coax Connector to Receptacle



RELATIVE RELIABILITIES		TEST TIMES (MINUTES)	
A	.3	1	6.0
B	.1	2	3.0
C	.2	3	10.0
D	.1	4	1.0
E	.3	5	7.0
	1.0		

$$E = R_A T_A + R_B T_B + R_C T_C + R_D T_D + R_E T_E$$

$= .3(4) + .1(8) + .2(14) + .1(14) + .3(8)$

$= 8.6$ minutes, expected fault isolation time

Figure 5. Simple Fault Isolation Tree

$$E = \Sigma R_i \quad t_i$$

where E is Expected fault isolation time

R_i is the Reliability of element i

t_i is the time to isolate element i, that is, the sum of all test times in the branch terminating at element i.

Thus, in Figure 5, the expected isolation time is 8.6 minutes.

The three variables, considered in various combinations by the strategies, are test power, test performance time, and element reliability. One strategy considers all three of these, and produces a fault isolation procedure (tree) which is optimal, i.e., the expected fault isolation time is minimal.

This strategy is determined by computing, at each stage in the troubleshooting process, that test which provides the maximum information per unit time. Information is computed according to Bayes theorem as the reduction in the total system uncertainty, i.e.

$$\Delta U = \Sigma p_i \log_2 p_i - \Sigma p_i' \log_2 p_i'$$

where ΔU = uncertainty reduction

p_i = probability of ith malfunction prior to test

p_i' = probability of ith malfunction after test

In general, this algorithm may not yield a true minimum, as the stepwise process does not consider the characteristics of the fault areas discriminated at each stage. A dynamic programming formulation was implemented to compute a true minimum. This process essentially "looks ahead", down each branch of the fault isolation tree, and is able to generate a slightly more efficient strategy. In one application of the Bayesian process the expected troubleshooting time for a system was 11.702 minutes, whereas the dynamic programming process yielded 11.568 minutes. If this close correspondence between results holds up for other systems, we will employ the Bayesian processor to estimate the optimum, as it is a rapid computation compared to the heavy computation load of dynamic programming.

It must be emphasized that the compute load to generate the optimum used here was not considered by the processor itself, i.e., the definition of optimality does not embrace time invested in producing the result. Human performers, on the other hand, seem to be quite sensitive to the time costs associated with planning their performance. Field troubleshooters have at times been criticized for performing tests when more planning and analysis seemed more productive. Whether or not maintainers tend to "under-plan", it is important to distinguish between machine computed solutions, and those developed in real time by human maintainers who forego manual performance to conduct cognitive tasks.

At the opposite extreme is a strategy in which tests are selected at random from the set of all tests which can offer any information about the status of the system. The random strategy which selects only productive tests provides an upper limit on rational troubleshooting time. Strategies can be formulated which are even less effective than random selection. The mean of the distribution of expected isolation times produced by random test selection, however, represents the time expected when no information is utilized for test selection except the results of previous tests.

Between the optimal strategy and the random strategy (on the dimension of effectiveness) lie six rational, suboptimal approaches, each of which considers one or two of the three variables used by the optimum strategy. A brief summary of all eight strategies follows (also see Table 2).

1. Optimum test selection. Tests are selected to minimize total expected isolation time. This strategy considers the time costs of the tests, the power of the tests, and the relative reliabilities of the system elements.
2. Element half-splitting, per unit time. Tests are selected to best split the suspected elements into two subsets of equal size, per unit time. This is strategy 1 with initial element reliabilities ignored.
3. Briefest test selection. The briefest test which can provide any information is selected at each stage. Only time cost is considered in the selection.
4. Half-splitting by reliability. Tests are selected to best split the suspected elements into two subsets of equal failure probability. This is strategy 1 with test time cost ignored.
5. Half-splitting by element. Tests are selected to best split the suspected elements into two subsets of equal size. This is equivalent to strategy 2 with test time cost ignored.
6. Check least reliable element, per unit time. Tests are selected to monitor the greatest probability of failure per unit time. Test time cost and element reliability are considered.

		VARIABLE CONSIDERED		
NO.	STRATEGY	TEST TIME	TEST POWER	RELIA-BILITY
1	Optimum Test Selection	YES	YES	YES
2	Element Half-Splitting, Per Unit Time (ignore element reliabilities)	YES	YES	NO
3	Briefest Productive Test Selection	YES	NO	NO
4	Half-Splitting by Reliability (ignore test time cost)	NO	YES	YES
5	Half-Splitting by Element	NO	YES	NO
6	Check Least Reliable Element, Per Unit Time (ignore test power)	YES	NO	YES
7	Check Least Reliable Element	NO	NO	YES
8	Random Test Selection	NO	NO	NO

Table 2. Eight Generic Fault Isolation Strategies

7. <u>Check least reliable element</u>. Tests are selected to check the least reliable elements first. Only reliability is considered in the selections.

8. <u>Random test selection</u>. Tests are selected at random (no repeating) without regard to test time cost, test power, or reliabilities.

These eight strategies were applied to a microcomputer system consisting of mainframe, video terminal, hardcopy printer, and disk drive unit (Figure 6). The representation of the system is shown in Figure 7. The results of the analysis, summarized in Table 3, will ultimately be evaluated in terms of experimentally observed maintenance performance on the system.

The relationships among the various fault isolation methods, however are interesting in their own right. The simple strategy of performing the briefest productive test (strategy 3) yielded an expected isolation time of 13.5 minutes, surprisingly close to the 11.7 minute optimum. Strategy 2, which uses test power and test cost, yielded 13.2 minutes expected isolation time, indicating that initial reliability data contributed little to the solution. The classical half-splitting strategy (perform a test to split the system in two) yields 21.3 minutes, whereas half-splitting into two equally reliable subsets (strategy 4) requires less time at 16.8 minutes.

The two strategies which emphasize checking unreliable elements perform poorly, at over forty minutes. These results are surprisingly close to random test selection (Figure 8), which yields a mean expected repair time of 49.7 minutes (N = 800).

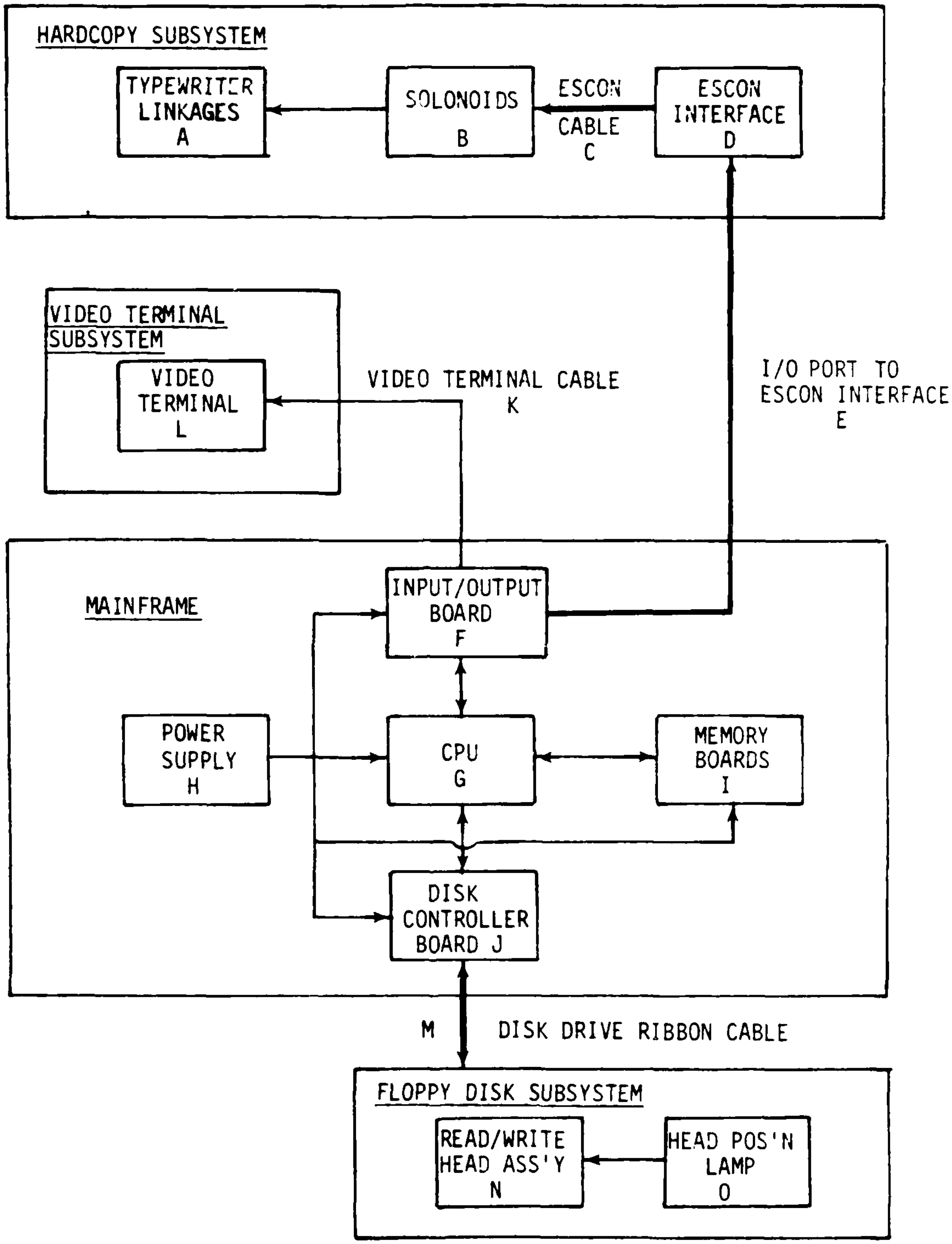


Figure 6. Microcomputer Block Diagram

	REPLACEABLE UNITS															
TEST	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	TEST TIME (MIN.)
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7.5
2	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	4.0
3	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	10.0
4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	12.0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	9.0
6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	7.0
7	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	10.0
8	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	15.0
9	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	55.0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	13.0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	5.5
12	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2.0
13	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	1.0
14	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	3.5
15	0	0	0	0	0	0	1	1	1	1	0	0	0	1	1	1.0
16	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	1.5
17	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	6.0
18	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	11.0

- - RELATIVE FAILURE PROBABILITIES - - - -

.219	.146	.007	.037	.007	.055	.044	.031	.073	.088	.007	.023	.007	.146	.110

Figure 7. Representation of Microcomputer System

	STRATEGY [1]							
ELEMENTS [2]	1	2	3	4	5	6	7	8
A	13.5	20.5	20.5	8.5	23.2	7.5	7.5	NOT COMPUTED
B	18.0	18.0	18.0	20.5	27.0	28.5	19.5	
C	29.5	29.5	29.5	35.5	23.2	71.5	125.5	
D	18.0	18.0	18.0	25.5	27.0	66.5	107.5	
E	20.5	13.0	13.0	42.5	19.0	78.5	116.5	
F	12.0	12.0	12.0	31.5	22.0	62.5	102.5	
G	9.0	9.0	9.0	8.0	8.0	54.0	105.0	
H	5.5	5.5	3.5	32.5	32.5	54.0	105.0	
I	2.0	2.0	2.0	32.5	32.5	114.5	92.5	
J	9.0	9.0	9.0	8.0	8.0	11.0	37.5	
K	27.0	27.0	27.0	46.5	37.0	159.5	124.0	
L	27.0	27.0	27.0	46.5	37.0	159.5	124.0	
M	29.5	29.5	29.5	42.5	19.0	89.5	125.5	
N	7.5	7.5	9.0	10.3	16.0	51.5	32.5	
O	7.5	7.5	9.0	10.3	16.0	16.5	34.5	
EXPECTED TIME	11.7	13.2	13.5	16.8	21.3	43.1	46.9	49.7

[1] See Table 2

[2] See Figure 6

Table 3. Element Isolation Times (Minutes) for Eight Generic Strategies

```
 Cost   Prop.
  15.27 0.012 |*****
  21.00 0.051 |*********************
  26.73 0.108 |*******************************************
  32.46 0.140 |********************************************************
  38.19 0.163 |*****************************************************************
  43.92 0.102 |*****************************************
  49.65 0.085 M**********************************
  55.38 0.059 |************************
  61.11 0.045 |******************
  66.84 0.031 |*************
  72.57 0.030 |************
  78.30 0.044 |******************
  84.03 0.030 |************
  89.76 0.040 |****************
  95.49 0.021 |*********
 101.22 0.015 |******
 106.96 0.011 |*****
 112.69 0.005 |**
 118.42 0.005 |**
 124.15 0.001 |*
 129.88 0.001 |*


800 trials,  mean cost=     49.68 variance=   525.604 std=    22.926
mininum cost=     15.27 max cost=    129.88
ONLY PRODUCTIVE TESTS USED.
```

Figure 8. Distribution of Fault Isolation Times With Random Test Selection (2 fault trees per *)

Examination of Table 3 reveals that the rank-order of fault isolation times for individual faults are relatively consistent across strategies. Those approaches which ignore test time cause the greatest departures from this tendency, since they may call for performing lengthy tests to check just a few unreliable elements.

The results of this one analysis certainly do not constitute a basis for generalization. Since the optimum strategy provides a true baseline of expert performance, it may prove to correlate best with observed maintenance activity, across different systems. If maintainers are generally parsimonious with time but not particularly prone to consider test power, then we may find actual maintenance performance resembles that of strategy 3. If, instead, maintainers focus their attention on unreliable elements, then we might expect performance more like strategy 7. And, if maintainers switch among time-dominant, reliability-dominant, and test power-dominant strategies, we might expect some function of strategies 3, 5 and 7 to provide a projection of maintenance workload. For example, if there is a tendency to select quick and easy tests early in a problem, and later shift to an enumerative search process as the possible faults emerge, we may employ strategies 3 and 7 to project the performance. Experimentation is needed to determine if such shifting strategy techniques are used by maintainers, and if so, to determine when and under what conditions in a fault isolation task such shifts will occur.

The most intriguing result of this one application is that the fault isolation performances and times were relatively constant across the time-dominant strategies and relatively constant at a higher level across the two reliability-dominant strategies. This suggests the interesting and very tentative hypothesis that the work required to isolate a particular fault may be highly determined by the design and less sensitive to individual differences of isolation method. Further application and experimentation are needed to test these early impressions.

VI. Experimentation in Maintainability Research

The conditions required to experimentally observe realistic maintenance performance are numerous and not readily achieved. While a number of interesting effects may be studied in a highly sanitized setting, the major problems confronting a maintainer may be lost in the process. High fidelity of field maintenance conditions is extremely difficult to attain while simultaneously capturing desired performance data. Today, the computer offers an attractive mechanism for tirelessly interacting with subjects and recording detailed performance data. The elegance of the data collection mechanism, however, must not require that the maintenance task be converted into a man-computer interaction task.

Particular experimental requirements will affect the types and extent of fidelity required or justified. The considerations may be classified into three categories: problem fidelity, performance fidelity, and environmental fidelity.

Problem Fidelity

Experimentation which addresses how maintainers generate their performance will usually be concerned with preserving, in the laboratory, the same problems faced by the maintainer in the field. In addition to the problem of identifying a possible fault, the real world troubleshooter faces uncertainties regarding (1) the current existence of a failure, (2) the current structure of the system, and (3) the accuracy of symptoms obtained.

Failure uncertainty. A maintainer who is assured that a system contains a persisting, catastrophic fault faces a different, and considerably simpler, problem than one operating in normal field conditions. The field maintainer must consider that no fault exists at all, or that the fault may be intermittent, marginally observable, or observable only in highly constrained configurations.

Under these conditions, if a test yields a normal result, the system elements involved in producing that indication may be provisionally considered as operational. If later results seem to conflict with this conclusion, then previous conclusions are suspect. In these cases, the real maintainer faces a difficult memory and logical problem in keeping track of what evidence is firm and what is suspect. When the possibilities of intermittent failures are considered by the maintainer, normal test results may have to be greatly discounted to avoid eliminating the true fault from suspicion.

Unfortunately, intermittant faults are not at all uncommon. In addition, numerous situations can create seeming intermittency even though the fault may be stable. The maintainer may observe different symptoms for a repeated performance of a test, yet not be able to ascertain if all aspects of the test were replicated. This is especially common when multiple sensors or external signals are used in the test. If the foregoing difficulties are artificially avoided in an experimental setting, normal results conclusively eliminate from suspicion all involved elements. Fault identification can therefore proceed in a manner which is not representative of field conditions.

Structural uncertainty. Most troubleshooting experimentation has been conducted in an environment of certainty regarding the structure of the system to be diagnosed. Typically, subjects are provided diagrams representing the structure of the system. Frequently, these are at the level of "signal flow" diagrams which represent simple connectivity of elements.

In the real world the maintainer confronts a somewhat different problem. First, real systems may be configured, via cables and switches, into a vast number of modes of operation. Many of these may depend upon conditions at remote locations which cannot be verified with ease or certainty. Secondly, the malfunction itself often has the effect of altering the system structure radically. Open or shorted leads do this, as well as some types of degradation or catastrophic failure of components. A further complication is introduced when systems normally alter their structure over time. Many computer-synchronized systems shift functions and form many times per second.

Thus, the real world troubleshooter often faces a system whose structure is unknown, either during fault diagnosis or during the initial appearance of the malfunction. Laboratory experimentation will embrace this dilemma only if representations of the target system are offered as nominal characterizations of system structure, and not as guaranteed system connectivity.

The impact of the variable becomes evident when real systems are diagramed in "signal flow" form. As seen in Figure 9, the connectivity of the experimental microcomputer system is trivially simple, and troubleshooting a system which is no more or less than that diagram is also trivial. Yet the real equipment is not easy to diagnose, for subtle and more complex cause-effect relationships, not captured by the connectivity diagram, must be considered by the troubleshooter.

Symptom uncertainty. Test results received in the real world are sometimes incorrect, or incorrectly interpreted. The indicator or test equipment may be faulty; the technical documentation may be misleading, incorrect, or incomplete; or a correct indication may be erroneously interpreted. Some experienced technicians we have observed are so wary of these possibilities that they do not assess single test results. Instead, they collect several readings and then consider if the combination of results is meaningful and consistent.

As with failure uncertainty, the consequence which emerges from symptom uncertainty is that much information received must either be provisionally processed or simply stored for later assessment. In either case, cognitive effort must be devoted to reassessing past results as troubleshooting progresses. To retain this aspect of the maintainer's problem requires that subjects be advised that test results provided may be in error, and that technical documentation provided may be imperfect. We would anticipate that the mere presentation of these warnings would significantly degrade troubleshooting performance. To actually introduce such error into

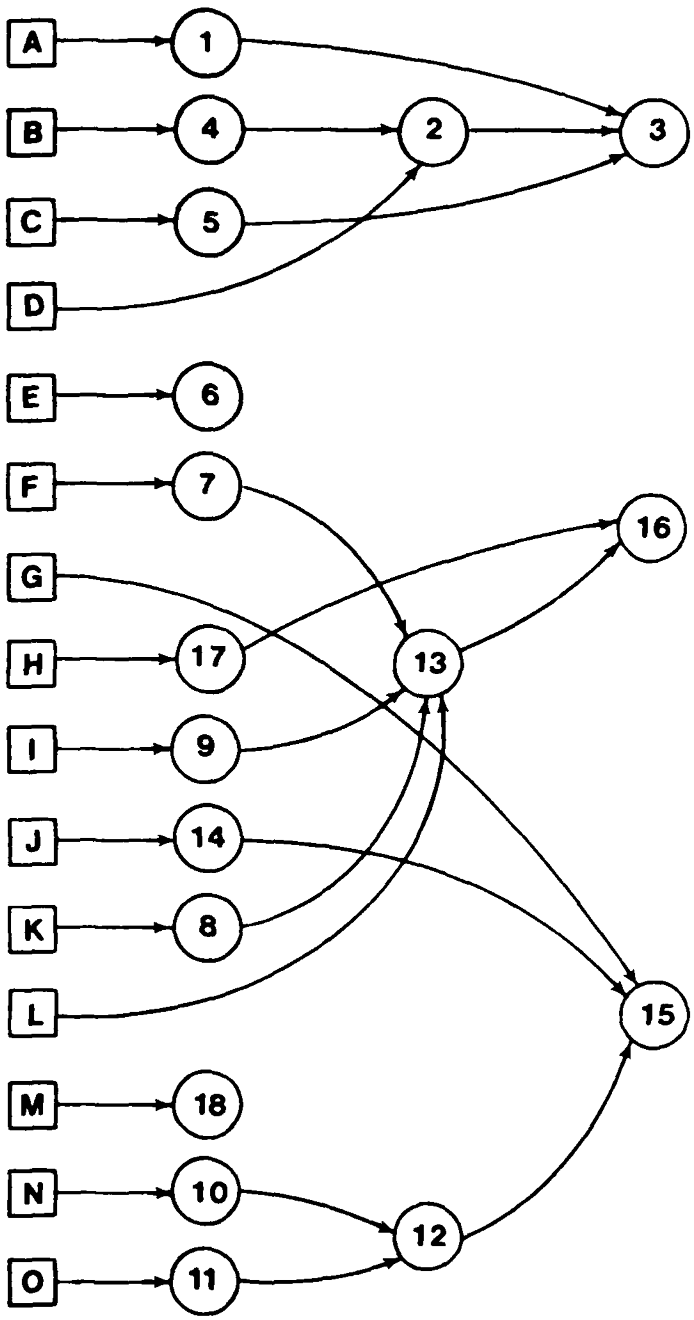


Figure 9. Test Dependency Network for Microcomputer System

test results or reference standards would further degrade performance.

A further interesting effect of symptom uncertainty is that abnormal readings must often be somewhat discounted, whereas normal readings may be more credible. For example, if a voltage of 16.5 is received at a test point which should read 16.7, the maintainer can feel relatively sure that the test equipment is operating and set up correctly. A reading of zero, however, could be obtained if the test equipment is not functioning or not set up properly.

We suspect that maintainers over-react to symptom uncertainties in the same way computer programmers often over-react to a hardware failure. Once a computer failure is encountered, programmers have considerable difficulty interpreting subsequent program bugs as such, for typically they have just invested great time and energy searching for a program bug which did not exist.

Performance Fidelity

The second component of experimental fidelity is related to the realism of performance which is allowed and required. Observed performance will be most representative of field performance if the subject operates in real time, receiving realistic sensory information, and is free to commit errors.

Real time performance. Actual maintenance is conducted in real time. Time devoted to cognitive activity (information utilization) is time which could be devoted to manual performance, and vice versa. There is strong evidence that maintainers somehow apply their cognitive time investment in a rational manner. Th y would rarely devote ten minutes to deciding which test to perf .m if they know of one or more tests which would require just a few minutes. Conversely, they would rarely make a snap judgement to initiate a long and arduous test procedure. Thus maintainers seem to be rationally parsimonious with their time resources by allocating cognitive time in relation to the consequences in performance.

Once a manual operation is in progress, the maintainer might perform a variety of cognitive processes including reviewing past results, planning for possible contingencies, and considering the possible sources of trouble. This cognitive activity, or possibly some new cues or information encountered during the operation, may cause the maintainer to terminate the task in progress and embark on a new course of action.

An experimental procedure which removes or distorts the time dimension can alter the process and the product of maintenance activity in unknown and, we suspect, profound ways. Typically, time costs of the possible alternatives are given to the subjects. These affect their decisions to some extent. Our pilot research employing this technique has convinced us, however, that subjects cannot accurately project, or imagine, the artificial time costs. Instead, they tend to select tests which minimize their actual time investment on the problem, rather than a computed, theoretical time score. In any case, subjects lose the opportunity to abort a test in progress as well as the opportunity to absorb information or to "think" while performing longer manual tests.

Information fidelity. It is possible that a considerable amount of the information used by a maintainer during fault isolation is not consciously or explicitly sought. Maintainers may depend on the visual appearance of the equipment to remind them of the testing options, the equipment functions, and system structure. They may discover clues, valid or not, while engaged in one activity, which cause them to initiate another. They may see, hear, smell, or feel aspects of the equipment which are unexpected. They may also take in and utilize the absence of symptoms which they might not think to explicitly sample in an experimental setting.

To alter this environment to one in which the maintainer senses only what he requests, is to create a substantially different information flow. At a minimum, the visual and auditory information should be realistic and complete.

Error fidelity. Depending upon the objectives of the experiment, the opportunities to commit performance errors may be either retained or eliminated. If manual performance errors are to be allowed, the subject must operate upon some real hardware. If errors are not to be considered, the subject either must not touch real hardware or else some error monitoring scheme must be employed.

The major difficulties which arise from use of actual hardware are (1) danger to subjects must be eliminated, (2) means for recognizing and recording performance elements must be developed, and (3) hardware must be periodically refurbished, both to maintain reliability and to remove visual clues to subjects.

For experimentation in electronics maintenance, development of custom hardware is most attractive. Safe and adequately complex systems can be configured from economical and low-power integrated circuits. Use of sockets and wire-wrap leads avoids soldering during component replacement, thus precluding subject injury as well as facilitating periodic refurbishing of the experimental vehicle. This approach also facilitates study of design alternatives, whereas existing operational hardware is difficult to modify for this purpose.

Instrumentation for sensing and recording performance data may either be built into the experimental vehicle, or it may be external to it. Built-in sensors could reliably detect switch changes and test point usage. Sensing visual monitoring of indicators would require installation of ancillary push buttons, activated by the subject, to check an indicator. While the sensors for switches and indicators could be somewhat standardized, very special techniques would be required to capture disassembly, adjustment, replacement, and reassembly performance data.

An economical and reliable alternative to use of built-in sensors is to employ video tape to record performance data. While somewhat inelegant by today's fully automated standards, video tape provides a verifiable, low cost record of performance. This approach does introduce a process involving reduction of taped content to digital form by human review. This can be facilitated by viewing the tape under computer control. Upon encountering a subject action, the analyst can press a key (on the computer keyboard) which stops the tape playback and automatically notes the frame number (30 or 60 frames per second, depending on the video equipment). When an identification code is entered for the action, the program computes the real time of the event and records the event digitally on disk.

To sense visual actions by external means would involve very expensive instrumentation. While technology has been developed to do this, it could be exceedingly difficult to implement and maintain. It seems reasonable, therefore, to utilize built-in push buttons, as described above, to mark each visual indicator check.

Generalizations regarding experimentation are more difficult in non-electronic domains. Existing operational equipment may be so large or expensive that only simulation can prompt representative performance in the laboratory. In other cases, generic mock-up may be necessary. In any case, the use of video tape for recording observed troubleshooting performance remains a viable technique which can remain relatively independent of the hardware employed in experimentation.

Environmental Fidelity

Maintenance in the services is often performed under challenging environmental conditions. Extreme temperature, poor lighting, confined space, high noise levels, and instability are just a few of the physical difficulties of restoring equipment in the field. Moderate environmental conditions can slow performance pace considerably. Extreme conditions can affect the work content itself.

The psychological factors in the field are significant as well. The rewards, penalties and fears associated with field maintenance may have considerable impact on performance.

The manner and extent to which maintenance performance is affected by these factors is not well established. Furthermore, the interactions between design and environment are not clear. We suspect that environment affects performance significantly and differentially (over designs), but that measures of relative merit of designs would be reasonably reliable under moderate conditions.

Research Plan

The initial experimentation planned will focus on the content of generated status identification (diagnostic test) sequences. Our objective is to find a basis for predicting performance of maintainers in acquiring new facts from manual and perceptual actions, and for predicting how these maintainers make cognitive use of acquired facts to direct subsequent activities. Initially, effects of errors in performance, manual performanc rate, and environment will be excluded from consideration. A computer-controlled video tape testbed has therefore been developed which displays correct performance of tasks chosen by the subject.

The testbed system is first used to present to a subject a qualitative description of the operation of the target system and the functional relationships among its components. This is done by means of a video-tape presentation of the system along with accompanying text displayed on the computer CRT. In a similar manner, the subject is next shown each of the diagnostic test and replacement procedures performed in real time with an accompanying explanation of their diagnostic function. At this point, a subject's understanding of the system and its associated tests may be examined and/or the subject could be allowed to review any segments of the preceding presentation which are not clear.

Following completion of the instructional phase of presentation, the subject is then ready to tackle some troubleshooting problems on the target system. At the beginning of each problem, the subject is presented with some very limited data about failure symptoms. From this point on the subject is free to "perform" any test or replacement that is deemed useful to correctly diagnose and repair the defective target system.

To perform a test, the subject presses a key associated with that test. The computer then determines what outcome the simulated malfunction would produce, positions the video tape unit to the segment showing that outcome, and plays the taped segment showing a technician performing the test and receiving a result. To disassemble, replace components, or reassemble, the subject presses a key associated with the action desired. Again, he views a taped segment showing that work. The subject may decide to reconfigure the system, swap cables, use test equipment, run diagnostic programs, and perform a number of operational tests, some of which involve partial disassembly. At any time, the subject may terminate work in progress by pressing a particular key.

Work proceeds in this way until the simulated equipment, the microcomputer system of Figures 6 and 7, is restored. This experimental technique meets most of the requirements of real time fidelity and information fidelity, and purposely precludes the possibility of performance errors. The subject observes rather than performs the selected actions, but retains the opportunity to terminate any action in progress. The visual and auditory information received is highly realistic, preserving the opportunity to pick up valuable incidental information while a test is being performed. For example, while a test is being conducted, an observant subject might notice some aspect of system configuration that could suggest a particularly fruitful next test to perform.

This testbed system appears to provide a useful experimental tool for analyzing aspects of maintenance performance while avoiding the requirement for manual performance skills, with an attendant high probability of error. Using this experimental tool, we intend to look at aspects of maintenance performance such as the following:

1. How accurately is the relationship between tests and malfunctions represented by the maintainer? This can be assessed at any point in the task, beginning with completion of the instructional phase up to the conclusion of the troubleshooting problem.
2. How do specific features of system design, such as modularity, affect the content of the status identification sequences? By altering the design of the target system from that shown in Figures 6 and 7, we can manipulate a number of basic features of the system's construction. By comparing subject performance across such changes in system construction we can obtain direct experimental evidence about the effect of such design features on troubleshooting performance.
3. To what extent is the subject sensitive to incidental information which is available (visually or auditorily) during performance of actions not specifically intended to obtain that information? Are there specific system design features which impact this ability to pick up incidental information?
4. Is diagnostic efficiency affected more by design parameters than it is by individual differences in status identification

sequences? By having a record of the test and replacement steps we can look at the effect of strategy variation upon overall diagnostic efficiency.

5. What is the depth of planning which typifies a subject's performance? Do human troubleshooters tend to be one step planners in a fault diagnosis task? By augmenting our testbed methodology with a record of each subject's "thinking out loud" protocol we may obtain data bearing upon this issue.

6. What are the criteria used by a subject in deciding to perform some action? The augmentation to our testbed just mentioned could allow us to obtain evidence on the nature of the decision criteria used by our subjects for test selection.

VII. Summary and Conclusions

The tools which exist today for assessing the maintenance workload composed by a system design are not sensitive in ways which are useful during the design phase, nor do they yield a profile of the performance which is involved in the maintenance task.

A considerable portion of maintenance activity is predictable and quantifiable, using traditional work measurement techniques (micromotion analysis). These include fixed diagnostic procedures prescribed in technical documentation, and restoration (disassembly, repair/replace/adjust, and assembly) tasks which are highly constrained by the physical structure of the system. A general representation of the physical structure of the system is sufficient to specify what actions are necessary. A catalog of action times provides the basis for computing performance time.

The primary obstacle to synthesizing a representative distribution of maintenance action sequences lies with the variability of troubleshooting performance. A number of models have been developed which address troubleshooting specifically or problem solving in general. While the flexibility and intuitive reasonableness of these models are progressing, none seem sufficiently developed to be of practical use at this time for the purpose of generating representative fault isolation sequences.

A-I models of troubleshooting are applications of more general research in A-I which has focused on the design of intelligent machine problem solving systems. Typically, these models are developed from assumptions that the problem solver has (1) a rather extensive representation of the problem domain, (b) a hierarchical approach to planning, and (3) a rigid overall control structure. While such assumptions may be quite reasonable ones in terms of design considerations for constructing an intelligent machine problem solving system, these assumptions appear to be less reasonable as hypotheses about the way human troubleshooters perform their task. First, the human troubleshooter does not appear to be able to solve a problem by "running" a complex mental simulation of the target system (either forward or backward). Second, the human troubleshooter does not appear to devise elaborate, hierarchical plans for projected actions; at times, the decisions for projected actions may be based upon a consideration of only the next step in the attempted solution. Third, it appears that the troubleshooter has available a range of decision criteria for choosing a next step, each of which derives from distinct features of the underlying representation or from special features of newly obtained data. Consequently, the particular choice criterion for use at each point in the fault isolation task may remain constant throughout the task, or it may vary as a result of new information being obtained or a change in the way the problem is being represented. A good model of the maintainer should reflect this variety and flexibility of strategic processes in troubleshooting.

One type of A-I problem solving model which is particularly appealing in light of these comments is the Hearsay-II system (Erman et al., 1980). The Hearsay system can provide a framework for building a problem solving model which may be quite flexible, opportunistic, and data driven in its operation. And, since the system is really a general control structure within which most other problem solving models may be instantiated, it can be used to develop and compare the performance of a number of distinct approaches to performing a specific task. This powerful feature provides a direct and controlled method for performing comparative evaluations of the performance of different troubleshooting models. Another dividend of this feature is that it provides a framework within which to develop a single model of troubleshooting which is itself a combination of alternative problem solving techniques, with the flexibility to switch from one technique to the next as a task demands. Future research, beyond the scope of our current project, should be devoted to the development of a troubleshooting model using techniques taken from the Hearsay-II system, which more faithfully reflect the kinds of decision making and problem solving engaged in by real maintainers.

The approach described in this report is based upon a family of primitive troubleshooting strategies, each of which recognizes none, some, or all of the following variables: test time cost, test power, and reliability. The strategies range from an optimal approach which minimizes fault isolation time, to an approach in which tests are selected at random. Each of these eight strategies generates a unique fault isolation procedure (fault tree) when applied to a representation of the design. The expected time to isolate a fault according to any of the resulting trees reflects the ease of performing the required tests and their power in revealing the internal state of the system.

Troubleshooting sequences performed in the field reflect the impact of numerous factors. At times, the environmental difficulties may override all other technical considerations in conducting fault isolation; avoidance of danger, pain, and serious errors may determine the nature of the performance. In a moderate environment, however, the maintainer is affected by the demands the system design places upon his abilities, and the opportunities the design affords him in locating the fault.

At some times, in a problem, the maintainer may be primarily concerned with quickly building up a symptom pattern in order to either identify the fault or to direct a more time consuming and focused search. At other times, the maintainer may be primarily concerned with checking a suspected element or function.

This research is based upon the hypothesis that one or more of the eight primitive fault isolation algorithms will reflect the impact of design in ways similar to experimentally observed performance. Experimentation will be conducted in which subjects will troubleshoot faults in a microcomputer system. A computer-controlled video tape system will respond to each subject decision, showing a real-time enactment of each test, including disassembly for access if necessary, and an enactment of each replacement, repair or adjustment. This is comparable to the subject directing another technician who carries out the decisions of the subject.

This experimental procedure eliminates the possibility of a manual performance error, yet imposes no restrictions on the fault isolation process employed by the subject. Furthermore, it retains the real time nature of troubleshooting, thus allowing subjects to abort an action in progress, and to "think" during performance of an action. Most importantly, time expended in cognitive activity accumulates with time expended performing observable work as it does in the real world.

The eight primitive fault isolation methods have been applied to the microcomputer system to be restored by experimental maintainers. For this system, the method which selects tests based on performance time alone produced near-optimum results. The methods which based test selection on their ability to check individual suspected (less reliable) elements, produced results nearly as poor as random test selection. There are reasons to believe that the shortest-test-first approach may be very efficient across designs in general. Furthermore, this is an easy and natural process to employ. If this effectiveness holds up across designs, we would expect it to be an effective predictor of human performance, at least for the early stages of a troubleshooting problem. An interesting training implication also would result, i.e., troubleshooting effectiveness may be more sensitive to symptom interpretation skills than strategic and planning skills.

There are some important issues at which to direct future research on the relationship between equipment design and maintenance task performance. For example, little research has been conducted which looks directly at the way in which specific cognitive skills are involved in maintenance task performance, what the performance limits are for these skills within the context of this task, and how their use is impacted by various features of the maintenance task and the equipment design. Such issues are clearly essential to the goal of gaining a more detailed understanding of maintenance and troubleshooting activities. As another example, the issue of manual performance error needs to be addressed. This may be accomplished by expanding the present research approach to one using real equipment in a controlled setting. We need to know how the maintainer's level of ability in performing the manual aspects of the maintenance interacts with the ability to efficiently perform the diagnostic and other cognitive aspects of the task. One possibility is that, as these skills become well learned (i.e., automated), concomitant diagnostic activity, such as symptom interpretation, becomes more efficient (e.g., more accurate).

Very few solid facts are presently established which help to clearly characterize the maintainer and how performance of his or her task can be related to properties of the system on which this craft is being performed. In the preceding discussion, we have considered this topic from many quite diverse perspectives. Using methods which draw upon the best features of all of these viewpoints, we hope to answer some of these questions and add some clarity to the currently rather blurred picture of the maintainer.

References

Amarel, S. On representations of problems of reasoning about actions. In D. Michie (Ed.), Machine Intelligence 3. New York: American Elsevier, 1968.

Balogh, F.S., Hennessy, J.F., & Reynolds, D.E. A comparison of demonstrated and achieved equipment maintainability. In Annals of Assurance Sciences, Proceedings of Annual Reliability and Maintainability Symposium, 1974.

Boecek, G.P. & Veitengruber, J.E. Human factor guidelines for caution and warning systems. Document No. D6-44200, Boeing Commercial Aircraft Company, November 1976.

Brown, Allen. Qualitative knowledge, causal reasoning, and the localization of failures (Technical Report AI-TR-362). Cambridge, Massachusetts: Massachusetts Institute of Technology, Artificial Intelligence Laboratory, March, 1977.

Chapman, L.J. & Chapman, J.P. Atmosphere effect re-examined. Journal of Experimental Psychology, 1959, 58, 220-226.

Cooper, G.E. A summary of the status of and philosophies relating to cockpit warning systems. NASA Contractor Report NAS2-9117, June, 1977.

deGroot, A.D. Perception and memory versus thinking. In B. Kleinmutz (Ed.), Problem Solving. New York: Wiley, 1966.

de Kleer, J. Causal and teleological reasoning in circuit recognition (Technical Report AI-TR-589). Cambridge, Massachusetts: Massachusetts Institute of Technology, Artificial Intelligence Laboratory, September, 1979.

de Valois, R.L. and de Valois, K.K. Spatial Vision. In M.R. Rosenberg and L.W. Porter (Eds.) Annual Review of Psychology, Vol. 31, 1980.

Duncker, K. On problem solving. Psychological Monographs, 1945, 58, No. 270.

Erman, L.D., Hayes-Roth, F., Leuer, V.R., & Reddy, R. The Hearsay-II speech understanding system: Integrating knowledge to resolve uncertainty. Computing Surveys, 1980, 12, 213-253.

Fikes, R.E. & Nilsson, N.J. STRIPS: A new approach to the application of theorem proving to problem solving. Artificial Intelligence, 1971, 2, 189-208.

Goldstein, I.P. Understanding simple picture programs (Technical Report 294). Cambridge, Massachusetts: Massachusetts Institute of Technology, Artificial Intelligence Laboratory, 1974.

Guiascu, S. Information Theory with Applications. New York, McGraw-Hill International, 1977.

Haber, R.N. Visual perception. In M.R. Resenzweig and L.W. Porter (Eds.) Annual Review of Psychology, Vol. 29, 1978.

Hayes-Roth, B. Human planning processes. Technical Report R-L670-ONR. Rand Corporation, Santa Monica, California, December, 1980.

Hayes-Roth, B., & Hayes-Roth, F. A cognitive model of planning. Cognitive Science, 1979, 3 (4), 275-310.

Hayes-Roth, B. & Thorndyke, P.W. Decision making during the planning process. Rand Technical Report N-1213-ONR, Santa Monica, California, October, 1980.

Henle, M. On the relationship between logic and thinking. Psychological Review, 1962, 69, 366-378.

Horne, R.C. Investigation of the Effects of Different Maintenance Procedures on the Reliability and Maintainability of the AN/ARC-27. Washington, D.C.: ARINC Corporation, 1962.

Horvath, W.J., Jr. Applicability of the Lognormal Distribution to Servicing Times in Congestion Problems. Operations Research Society, Jan-Feb 1959 Meeting.

Karger, D.W. & Bayha, F.H. Engineered Work Measurement, Industrial Press, Inc., New York, 1966.

Kahneman, D. & Tversky, A. Prospect Theory: An analysis of decision under risk. Econometrika, 1979, 47, 263-291.

Leuba, H.R. The Symptom Matrix. Washington D.C.: ARINC Research Corporation, 1962. (Publ. No. 4607-1-294).

Maier, N.R.F. Reasoning in humans II. The solution of a problem and its appearance in consciousnesss. Journal of Comparative Psychology, 1931, 12, 181-194.

Mayer, R.E. & Revlin, R. An information processing framework for research on human reasoning. In R. Revlin and R.E. Mayer (Eds.), Human Reasoning. New York; Wiley, 1978.

Miller, G.A. The magical number seven plus or minus two: Some limits on our capacity for processing information. Psychological Review, 1956, 63, 81-97.

Moray, N. Attention: Selective Processes in Vision and Hearing. New York: Academic Press, 1970.

Moray, N. The role of attention in the detection of errors and the diagnosis of failures in man-machine systems. In J. Rasmussen & W.B. Rouse (Eds.), Human Detection and Diagnosis of System Failures. New York: Plenum Press, 1981.

Munger, M.R. & Willis, M.P. Development of an Index of Electronic Maintainability--A Research Report. Fort Monmouth, New Jersey, U.S. Army Signal Support Agency, 1959.

Mynott, C.R., Doherty, M.E., & Tweney, R.D. Confirmation bias in a simulated research environment: An experimental study of scientific inference. Quarterly Journal of Experimental Psychology, 29, 1977, 85-95.

Nauta, F. & Bragg, L. Army electronic equipment support options: Assessment of maintenance complexity. Working Note ML904-2, November 1980, Logistics Management Institute, Washington, D.C.

Neisser, U. Cognitive Psychology. New York: Appleton-Century-Crofts, 1967.

Newell, A. & Simon, H.A. GPS, a program that simulates human thought. In E. Fergenbaum & J. Feldman (Cdr.) Computer and Thought. New York: McGraw-Hill, 1963.

Newell, A., & Simon, H.A. Human Problem Solving. Englewood Cliffs, New Jersey: Prentice-Hall, 1972.

Nilsson, N.J. Some examples of A-I mechanisms for goal-seeking, planning, and reasoning. In F. Klix (Ed.), Human and Artificial Intelligence. New York: North-Holland, 1979.

Nilsson, N.J. Principles of Artificial Intelligence. Palo Alto, California: Troza, 1980.

Norman, D.A. Memory and Attention. (2nd Ed.) New York: Wiley, 1976.

Norman, D.A. Slips of the mind and an outline for a theory of action. Center for Human Information Processing, Report No. 7905, University of California at San Diego, November, 1979.

Norman, D.A. Errors in human performance. Center for Human Information Processing, Report No. 8004, University of California at San Diego, August, 1980.

Norman, D.A. & Bobrow, D.G. On data-limited and recourse limited processes. Cognitive Psychology, 1975, 7, 44-64.

Pau, L.F. Application of pattern recognition to failure analysis and diagnosis. In J. Rasmussen and W.B. Rouse, (Eds.) Human Detection and Diagnosis of System Failures. New York: Plenum Press, 1981.

Pezolli, J.A., & Frase, L.T. Mediated facilitation of syllogistic reasoning. Journal of Experimental Psychology, 1968, 78, 228-232.

Raphael, B. Research and applications - Artificial Intelligence, Stanford Research Institute Final Report on Project 8973, Advanced Research Projects Agency, NASU-2164, December, 1971.

Rasmussen, J. & Rouse W.B. (Eds.) Human Detection and Diagnosis of System Failures. New York: Plenum Press, 1981.

Revlis, R. Syllogistic reasoning: Logical decisions from a complex data base. In R. Falmagne (Ed.) Reasoning: Representation and Processes in Children and Adults. Hillsdale, New Jersey: Erlbaum, 1975 (a).

Revlis, R. Two models of syllogistic reasoning: Feature selection and conversion. Journal of Verbal Learning and Verbal Behavior, 1975, 14, 180-195, (b).

Rigney, J.W., Cremer, R.H., Towne, D.M. & Mason, A.K. A computer program for generating inherent corrective maintenance workload times. Behavioral Technology Laboratories, Technical Report No. 50, November 1966, University of Southern California.

Rigney, J.W., Cremer, R.H., & Towne, D.M. Measurement and prediction of cognitive loadings in corrective maintenance tasks: I. A bayesian approach. Behavioral Technology Laboratories, Technical Report No. 46, February, 1966, University of Southern California.

Rouse, W.B. & Rouse, S.H. Measures of the complexity of fault diagnosis tasks. IEEE Transactions on Systems Man and Cybernetics, 1979, (11), 720-727.

Sacerdoti, E.D. A structure for plans and behavior (Technical Note 109). Menlo Park, California: Stanford Research Institute, August, 1975.

Siegel, A.I. & Wolf, J.J. Man-machine Simulation Models. New York: John Wiley, 1969.

Smith, R.J. High-cost Lemons in the U.S. Arsenal. Science, 212, 309-312, 1981.

Strieb, M.I., Glenn, F.A., & Wherry, R.J. The Human Operator Simulator: An Overview. Presented at the NATO AGARD Conference on Manned Systems Design, Freiburg, West Germany, September, 1980.

Towne, D.M. ARMAN - A computer system for applying MTM, Proceedings Conference of MTM Applications, Los Angeles, 1968.

Towne, D.M. New Software Program Aids Work Measurement of Clerical Operations, Industrial Engineering, Vol. 12, No. 8, August 1980.

Tversky, A. & Kahneman, D. Judgement under uncertainty: Heuristics and biases. Science, 1974, 185, 1124-1131.

Wescourt, K.T. and Hemphill, L. Representing and teaching knowledge for troubleshooting/debugging (Technical Report 292). Stanford, California: Institute for Mathematical Studies in the Social Sciences, February, 1978.

Wohl, J.G. Diagnostic behavior, system complexity, and repair time: a predictive theory (Technical Report M80-0008). Bedford, Massachusetts: MITRE Corporation, April, 1980.

DISTRIBUTION LIST

1 CDR Paul R. Chatelier
Office of the Deputy Under
Secretary of Defense
OUSDRE (E&LS)
Pentagon, Room 3D129
Washington, D.C. 20301

5 Director
Engineering Psychology Programs
Code 455
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 Director
Manpower, Personnel and Training
Code 270
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 Director
Information Systems Program
Code 437
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 Special Assistant for Marine
Corps Matters
Code 100M
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 Commanding Officer
ONR Western Regional Office
ATTN: Dr. E. Gloye
1030 East Green Street
Pasadena, CA 91106

1 Dr. Robert G. Smith
Office of the Chief of Naval
Operations, OP 987H
Personnel Logistics Plans
Washington, D.C. 20350

1 Dr. Jerry C. Lamb
Combat Control Systems
Naval Underwater Systems Center
Newport, RI 02840

1 Director
Communication & Computer Technology
Code 240
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 Director
Tactical Development & Evaluation
Support- Code 230
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 Director
Operations Research Programs
Code 434
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 CDR. K. Hull
Code 410
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

1 Commanding Officer
ONR Western Regional Office
ATTN: Mr. R. Lawson
1030 East Green Street
Pasadena, CA 91106

6 Director
Naval Research Laboratory
Technical Information Division
Code 2627
Washington, D.C. 20375

1 Dr. W. Mehuron
Office of the Chief of Naval
Operations, OP 987
Washington, D.C. 20350

1 Naval Training Equipment Center
ATTN: Technical Library
Orlando, FL 32813

1 Human Factors Department
Code N215
Naval Training Equipment Center
Orlando, FL 32813

1 Dr. Gary Poock
Operations Research Department
Naval Postgraduate School
Monterey, CA 93940

1 Mr. Warren Lewis
Human Engineering Branch
Code 8231
Naval Ocean Systems Center
San Diego, CA 92152

1 Mr. Arnold Rubinstein
Naval Material Command
NAVMAT 0722 - Rm. 508
800 North Quincy Street
Arlington, VA 22217

1 Commander
Naval Air Systems Command
Crew Station Design
NAVAIR 5313
Washington, D.C. 20361

1 Commander
Naval Electronics Systems Command
Human Factors Engineering Branch
Code 4701
Washington, D.C. 20360

1 CDR Robert Biersner
Naval Medical R&D Command
Code 44
Naval Medical Center
Bethesda, MD 20014

1 Dr. George Moeller
Human Factors Engineering Branch
Submarine Medical Research Lab
Naval Submarine Base
Groton, CT 06340

1 Dr. James McGrath, Code 302
Navy Personnel Research and
Development Center
San Diego, CA 92152

1 Dr. Alfred F. Smode
Training Analysis and Evaluation
Group
Naval Training Equipment Center
Code N-00T
Orlando, FL 32813

1 Mr. H. Talkington
Ocean Engineering Department
Naval Ocean Systems Center
San Diego, CA 92152

1 Dr. A. L. Slafkosky
Scientific Advisor
Commandant of the Marine Corps
Code RD-1
Washington, D.C. 20380

1 Commander
Naval Air Systems Command
Human Factors Programs
NAVAIR 340F
Washington, D.C. 20361

1 Mr. Phillip Andrews
Naval Air Systems Command
NAVSEA 0341
Washington, D.C. 20362

1 Mr. Milon Essoglou
Naval Facilities Engineering
Command
R&D Plans and Programs
Code 03T
Hoffman Building II
Alexandria, VA 22332

1 Dr. Arthur Bachrach
Behavioral Sciences Department
Naval Medical Research Institute
Bethesda, MD 20014

1 Head
Aerospace Psychology Department
Code L5
Naval Aerospace Medical Research Lab
Pensacola, FL 32508

1 Navy Personnel Research and
Development Center
Planning & Appraisal
Code 04
San Diego, CA 92152

1 Navy Personnel Research and
Development Center
Performance Measurement &
Enhancement
Code 309
San Diego, CA 92152

1 Lt. D. McBride
Human Factors Engineering Division
Naval Air Development Center
Warminster, PA 18974

1 Human Factors Engineering Branch
Code 1226
Pacific Missile Test Center
Point Mugu, CA 93042

1 Human Factor Engineering Branch
Naval Ship Research and Development
Center, Annapolis Division
Annapolis, MD 21402

1 Mr. Merlin Malehorn
Office of the Chief of Naval
Operations (OP-115)
Washington, D.C. 20350

1 Dr. Joseph Zeidner
Technical Director
U.S. Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

1 Technical Director
U.S. Army Human Engineering Labs
Aberdeen Proving Ground, MD 21005

1 Human Engineering Division
USAF AMRL/HES
Wright-Patterson AFB, OH 45433

1 Dr. Kenneth Gardner
Applied Psychology Unit
Admiralty Marine Technology
Establishment
Teddington, Middlesex TW11 OLN
ENGLAND

1 CDR Norman Lane
Code 6021
Naval Air Development Center
Warminster, PA 18974

1 Human Factors Branch
Code 3152
Naval Weapons Center
China Lake, CA 93555

1 Human Factors Section
Systems Engineering Test
Directorate
U.S. Naval Air Test Center
Patuxent River, MD 20670

1 CDR W. Moroney
Code 55MP
Naval Postgraduate School
Monterey, CA 93940

1 Mr. J. Barber
HQS, Department of the Army
DAPE-MBR
Washington, D.C. 20310

1 Director, Organizations and
Systems Research Laboratory
U.S. Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

1 U.S. Air Force Office of Scientific
Research
Life Sciences Directorate, NL
Bolling Air Force Base
Washington, D.C. 20332

1 Dr. Earl Alluisi
Chief Scientist
AFHRL/CCN
Brooks AFB, TX 78235

1 Director, Human Factors Wing
Defence & Civil Institute of
Environmental Medicine
Post Office Box 2000
Downsview, Ontario M3M 3B9
CANADA

AD-A102 513 UNIVERSITY OF SOUTHERN CALIFORNIA REDONDO BEACH BEHAV--ETC F/6 9/2
DESIGN FOR THE MAINTAINER: PROJECTING MAINTENANCE PERFORMANCE F--ETC(U)
JUL 81 D M TOWNE, M R FEHLING, N A BOND N00014-80-C-0493

UNCLASSIFIED TR-95 NL

2 OF 2
AD A
102513

END
DATE
FILMED
9-81
DTIC

12 Defense Technical Information Center
Cameron Station, Bldg. 5
Alexandria, VA 22314

1 Dr. Craig Fields
Director, Cybernetics Technology Office
Defense Advanced Research Projects Agency
1400 Wilson Blvd
Arlington, VA 22209

1 Dr. Lloyd Hitchcock
Federal Aviation Administration
ACT 200
Atlantic City Airport, NJ 08405

1 Dr. M. Montemerlo
Human Factors & Simulation Technology, RTE-6
NASA HQS
Washington, D.C. 20546

1 Dr. Robert R. Mackie
Human Factors Research, Inc.
5775 Dawson Avenue
Goleta, CA 93017

1 Dr. Jesse Orlansky
Institute for Defense Analysis
400 Army-Navy Drive
Arlington, VA 22202

1 Dr. T. B. Sheridan
Department of Mechanical Engineering
Massachusetts Institute of Technolgy
Cambridge, MA 02139

1 Dr. Arthur I. Siegel
Applied Psycological Services, Inc.
404 East Lancaster Street
Wayne, PA 19087

1 Dr. Harry Snyder
Department of Industrial Engineering
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

1 Dr. W. S. Vaughan
Oceanautics, Inc.
422 6th Street
Annapolis, MD 21403

1 Dr. Robert T. Hennessy
NAS - National Research Council
JH #819
2101 Constitution Ave., N.W.
Washington, D.C. 20418

1 Dr. M. G. Samet
Perceptronics, Inc.
6271 Variel Avenue
Woodland Hills, CA 91364

1 Dr. Robert Williges
Human Factors Laboratory
Virginia Polytechnical Institute and State University
130 Whittemore Hall
Blacksburg, VA 24061

1 Dr. Alphonse Chapanis
Department of Psychology
The Johns Hopkins University
Charles and 34th Streets
Baltimore, MD 21218

1 Dr. Elizabeth Kruesi
General Electric Company
Information Systems Programs
1755 Jefferson Davis Highway
Arlington, VA 22202

1 Dr. Ward Edwards
Director, Social Science Research Institute
University of Southern California
Los Angeles, CA 90007

1 Dr. James H. Howard, Jr.
Department of Psychology
Catholic University
Washington, D.C. 20064

3 Journal Supplement Abstract Service
American Psychological Association
1200 17th Street
Washington, D.C. 20036

1 Dr. Christopher Wickens
University of Illinois
Department of Psychology
Urbana, IL 61801

1 Dr. Babur M. Pulat
Department of Industrial Engineering
North Carolina A&T State University
Greensboro, NC 27411

1 Mr. Tim Gilbert
The MITRE Corporation
1820 Dolly Madison Blvd.
McLean, VA 22102

1 Dr. Lola Lopes
Department of Psychology
University of Wisconsin
Madison, WI 53706

1 Dr. Stanley N. Roscoe
New Mexico State University
Box 5095
Las Cruces, NM 88003

1 Dr. Rex Brown
Decision Science Consortium
Suite 721
7700 Leesburg Pike
Falls Church, VA 22043

1 Dr. Edward R. Jones
Chief, Human Factors Engineering
McDonnell-Douglas Astronautics
Company
St. Louis Division
Box 516
St. Louis, MO 63166

1 Dr. Richard W. Pew
Information Sciences Division
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, MA 02138

1 Dr. David J. Getty
Bolt Beranek & Newman, Inc.
50 Moulton Steet
Cambridge, MA 02138

1 Dr. A. K. Bejczy
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91125

1 Mr. Joseph G. Wohl
Alphatech, Inc.
3 New England Executive Park
Burlington, MA 01803

1 Dr. Wayne Zachary
Analytics, Inc.
2500 Maryland Road
Willow Grove, PA 19090

DATE
FILMED
8