ANCR-1273
UC-79h
DATE PUBLISHED—APRIL 1976

IDAHO NATIONAL ENGINEERING LABORATORY

A COLLECTION OF METHODS FOR RELIABILITY AND SAFETY ENGINEERING

PREPARED BY AEROJET NUCLEAR COMPANY FOR
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
IDAHO OPERATIONS OFFICE UNDER CONTRACT E(10-1) -1375

DISCLAIMER

Printed in the United States of America
Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22161
Price: Printed Copy $7.50; Microfiche $2.25

ANCR-1273

Distributed Under Category:
UC-79h
LMFBR - Structural
Material and Design Engineering
TID-4500, R64

A COLLECTION OF METHODS FOR RELIABILITY AND SAFETY ENGINEERING

BY

J. B. Fussell
D. M. Rasmuson
J. R. Wilson
G. R. Burdick
J. C. Zipperer

AEROJECT NUCLEAR COMPANY

Date Published – April 1976

PREPARED FOR THE
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
IDAHO OPERATIONS OFFICE
UNDER CONTRACT NO. E(10-1)-1375

GENERAL ACKNOWLEDGMENTS

We gratefully acknowledge J. H. Carlson, F. X. Gavigan, and W. H. Hannum for their numerous contributions that made this project possible. Also, we extend gratitude to P. M. Lang and R. R. Stiger for their efforts that influenced the results presented. We are grateful to C. F. Miller for preparing the figures for this document.

ABSTRACT

This document contains five reports each describing a method of reliability and safety engineering.

Report I provides a conceptual framework for the study of component malfunctions during system evaluations. Report II provides methods for locating groups of critical component failures such that all the component failures in a given group can be caused to occur by the occurrence of a single separate event. These groups of component failures are called common cause candidates. Report III provides a method for acquiring and storing system-independent component failure logic information. The information stored is influenced by the concepts presented in Report I and also includes information useful in locating common cause candidates. Report IV puts forth methods for analyzing situations that involve systems which change character in a predetermined time sequence. These phased missions techniques are applicable to the hypothetical "accident chains" frequently analyzed for nuclear power plants. Report V presents a unified approach to cause-consequence analysis, a method of analysis useful during risk assessments. This approach, as developed by the Danish Atomic Energy Commission, is modified to reflect the format and symbology conventionally used for other types of analysis of nuclear reactor systems.

CONTENTS

REPORT I

AN ANALYSIS OF CAUSES OF COMPONENT MALFUNCTIONS

J. B. Fussell

ABSTRACT

In the course of system reliability and safety analysis, special problems with regard to component malfunctions arise during treatment of maintained systems. This report is concerned with:

(1) Understanding the considerations that must be given to component malfunction.

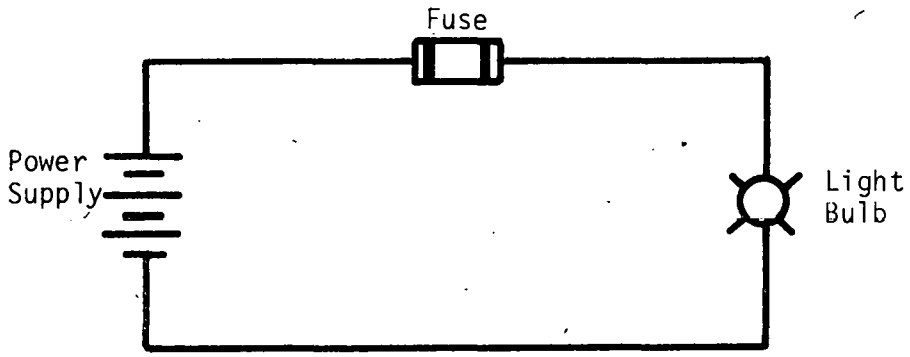
(2) Analyzing the general case quantitatively

(3) Introducing areas needing further investigation.

The report is written for the analyst with a basic knowledge of system reliability assessment techniques.

PRELUDE

The most advanced quantitative techniques have revealed inconsistent system reliability analysis results. These inconsistencies cannot be rationalized as being due to inadequate system logic models or errors in the input data. Anomalous results can be observed during analysis of even the following simple example system design.



An excessive number of fuse openings in similar systems has prompted a reliability analysis of the system to determine the expected fraction of the time the system will be inoperable due to fuse failures; that is, the asymptotic system unavailability is to be determined.

The analyst finds two sources of fuse failures, either of which will cause the fuse to open:

(1) The fuse opens because of defects, wearout, . . . that is, the fuse opens for reasons that are attributed to the fuse itself

(2) The power supply surges, resulting in an overcurrent through the fuse of sufficient magnitude to open the fuse.

The failure rate for Cause (1) is 10^{-6}/hr and since the fuse is located under several layers of shroud, the mean time to repair is 100 hr. The power supply output is controlled by a governor that fails with a failure rate of 10^{-2}/hr, allowing the generator to overspeed. One hour is required for the simple repair of the governor to restore correct generator output. Recognition that Cause (1) or Cause (2) will logically result in the fuse opening, use of standard methods for quantitative evaluation[1 through 9] and application of the assumption that Causes (1) and (2) are statistically independent result in the following:

$$\bar{a}_F = \bar{a}_1 + \bar{a}_2 - \bar{a}_1 \bar{a}_2$$

$$\bar{a}_F \approx 0.01$$

$$\lambda_F = \lambda_1 + \lambda_2$$

$$= 1.0001 \times 10^{-2}$$

and the mean time to repair

$$\tau_F = \frac{\overline{a}_F}{1\ \overline{a}_F}\ \frac{1}{\lambda_F}$$

$$= 1.01 \text{ hr}$$

where

$\overline{a}$ = the asymptotic unavailability ($\approx\lambda\tau$)

λ = failure rate

τ = mean time to repair

F = subscript denoting fuse malfunction characteristics for both of the causes

1 = subscript denoting characteristics due to Cause (1)

2 = subscript denoting characteristics due to Cause (2).

The appendix presents an explanation of the equations used for all the calculations given in this report. Since 100 hours are required to repair the fuse in any case, the 1.01-hour value of τ_F is ridiculous. The system asymptotic unavailability, as calculated correctly for this system by use of only Causes (1) and (2), is in fact slightly greater than 0.5, which is substantially different from the calculated 0.01.

This report is concerned with understanding the mechanisms that cause such erroneous results and how to work the problem correctly. The understanding of these mechanisms comes from a generic study of the manner in which components malfunction.

ACKNOWLEDGMENTS

Gratitude and acknowledgment are extended to Professor R. E. Barlow, University of California at Berkeley, for his comments that influenced this report. Also, G. R. Burdick, R. J. Crump, D. M. Rasmuson, M. E. Stewart, and J. R. Wilson are recognized for their suggestions.

CONTENTS FOR REPORT I

FIGURES

A COLLECTION OF METHODS FOR RELIABILITY AND SAFETY ENGINEERING

I. INTRODUCTION

The purpose of this report is to propose a conceptual framework for the study of component malfunctions during system reliability and safety analyses, as well as to propose a procedure for treatment of these malfunctions. Here a component malfunction is defined as an undesired output from the component with respect to the main system failure of interest. The concepts presented apply to all systems including abstract and hardware systems.

A system logic model, either expressed or implied, is required as a part of a system reliability or safety analysis. A fault tree is an example of such a logic model and in particular is a failure logic model. During the construction of a failure logic model, component malfunctions are invariably considered. These component malfunctions result from four possible sources:

(1) Primary failures

(2) Secondary failures

(3) Primary faults

(4) Secondary faults.

Failures, either primary or secondary, result in component malfunctions that require repair of the component before the malfunction is corrected. Primary faults and secondary faults result in component malfunctions that can be corrected without maintenance of the component in question. Repair is a reversal of basic event state from the failed state to the unfailed state.

Categorization of the sources of component malfunctions into primary or secondary causes is largely a matter of philosophy rather than definitive concepts. Primary causes, either failures or faults, result in component malfunction for which the component itself is held accountable. Secondary causes, either failures or faults, result in component malfunctions for which the component itself is not held accountable. This concept of accountability implies an envelope of conditions that constitute expected functional and environmental input. Causes that are a breach of this envelope are secondary causes. Since this envelope is seldom explicitly defined, primary or secondary cause classification is usually subjectively assigned by the analyst.

Specifically then, primary failures are causes of component malfunctions for which the component is held accountable and which require that the component be repaired

before the component malfunction is corrected. An example of a primary failure that causes a light bulb to malfunction is the filament opening during normal operation; that is, no overcurrent has been experienced, and so forth.

Secondary failures are causes of component malfunctions for which the component itself is not held accountable; however, the component must be repaired before the malfunction is corrected. Other repairs are also required to remove the sources of the secondary failure. Secondary failure causes can be generated within the system or can result from effects external to the system. An example of a secondary failure that causes an amplifier to malfunction is highly corrosive acid spilling from a broken pipe into the amplifier and causing the amplifier to fail.

Primary faults are causes of component malfunctions for which the component is held accountable; however, the component has the capability to carry out self-repair. At present this malfunction cause is used during analysis involving biological systems for which healing is possible. Hardware components with internal artificial intelligence also can have the capability for self-repair.

Secondary faults are causes of component malfunctions for which the component is not held accountable and the component, immediately or soon after the secondary fault sources are repaired, functions properly. Secondary fault causes can be generated within the system or can result from effects external to the system. An example of a secondary fault that causes automobile ignition breaker points to malfunction is water collecting between the breaker points and producing a short circuit across them. After the source of the water is removed and the points have had time to dry, the malfunction is corrected. No contact breaker repair is required.

The implication of the methods presented here with respect to logic model construction and analysis is given in Sections II and III, respectively. A quantitative treatment procedure is given in Section IV and sample problems are given in Section V. Section VI gives several comments with regard to dependency problems and, finally, in Section VII, conclusions are presented.

II. IMPLICATIONS WITH RESPECT TO LOGIC MODEL CONSTRUCTION

The need for detail when component malfunctions are treated during system reliability and safety analyses has long been recognized. D.F. Haasl introduced the concepts of primary and secondary failures to the literature in 1965[10]. Later Haasl propounded the concept of a "command fault", a special case of a secondary fault for which the system itself commands the component to malfunction. Specifically, a command fault is defined as a system-generated secondary fault such that the component immediately functions properly when the secondary fault sources are repaired.

Secondary and primary causes of component malfunction are treated conveniently with conventional fault tree symbols. However, the definition of one of the symbols needs to be updated. The circle symbol has frequently been said to represent primary failures. In practice, the circle is used to represent all sources of component malfunction that are not specifically indicated elsewhere. The circle is, then, a catchall symbol for system component failure causes and, therefore, represents the resolution of the analysis not only with regard to what entities in the system are considered components, but also with regard to the detail expressed for the component failure causes. As an illustration, if a fuse is selected as a component, then a circle is used to represent fuse failure causes. The resolution of the analysis is extended if particular secondary causes are noted and developed, in which case the fuse failure is developed by a logical OR gate with inputs consisting of the secondary causes and the circle symbol. If these causes are not to be developed, they are simply implied by the circle. The resolution of the analysis is also extended if the fuse parts are considered as the system components. The preferred connotation of the circle symbol is "basic event" rather than "primary event".

As a general rule, command faults are always logically developed. Because the source of the command fault is the system itself, failure to develop this fault during an analysis of the system is inconsistent. In fact, in some situations a component malfunction is developed by use of only a command fault. The circle symbol is not used. For example, if the component malfunction is a pump producing flow at the wrong time, then generally only a command fault is used for the logical development.

In summary, logical development of the secondary causes of component malfunctions is not necessarily required. However, the analyst should consider all the sources of these malfunctions for each pertinent system component and develop those that are deemed important. The subject of component malfunction sources is far from academic because the procedure for quantitative treatment depends explicitly on the types of secondary causes developed in the system logic model.

III. IMPLICATIONS WITH RESPECT TO QUALITATIVE AND QUANTITATIVE ANALYSIS

System logic models that contain secondary cause development, in general, require analysis techniques different from those that do not. The reason for the difference is that if a component malfunctions because of a secondary cause, repair of the basic events used for the logical development of the secondary cause does not result in immediate repair of the component that has malfunctioned. Command faults are an exception. The implication here is that a system logic model containing secondary failure cause development, other than command fault development, does not give "coherent" minimal cut sets.

A coherent set of minimal cut sets has a monotonically increasing structure function and all basic events are relevant. By the definition of a minimal cut set, all the basic events in a minimal cut set are relevant[3]. The set of minimal cut sets from a logic model of a maintained (repairable) system is monotonic if (a) occurrence of any basic event always increases the probability of occurrence of the main system failure of interest, called the TOP event, and (b) repair of any basic event always decreases the probability of occurrence of the TOP event. If a model contains secondary cause development, the probability that the TOP event occurs is not necessarily decreased when relevant basic events that have occurred are repaired; if that is the case, the model is not coherent. Barlow and Proschan,[11] present further information concerning coherent structures.

Qualitative analysis of logic models involves formulating conclusions based on the minimal cut sets obtained from the system logic model. Minimal cut sets obtained from a logic model containing developed secondary failures are conceptually different from those obtained from a logic model not showing secondary causes. Although in either case the TOP event occurs when any minimal cut set fails, the TOP event is not necessarily repaired when all the minimal cut sets from a logic model showing secondary cause development are repaired[a].

Quantitative analysis of logic models involves obtaining reliability characteristics for the TOP event from reliability characteristics of the basic events. All presently available techniques for quantitative evaluations assume the system logic model is coherent. Since logic models showing secondary causes are not generally coherent, no method is available for quantitative evaluation of these logic models. The remainder of this report is concerned with presenting a procedure for quantitative treatment of logic models that contain secondary cause development.

---

[a] A minimal cut set is failed when all of the basic events contained in that minimal cut set are failed. A minimal cut set is repaired when one or more of the basic events contained in that minimal cut set are repaired.

IV. QUANTITATIVE TREATMENT PROCEDURES

Quantitative treatment of coherent logic models, that is, logic models that do not contain secondary cause development, has been dealt with in detail in the literature[4 - 11] and will not be covered here. However, knowledge of these methods is helpful for the remainder of this paper. The appendix to this report presents the calculational technique used here.

The general approach used to evaluate system logic models that show secondary cause development is to transform the logic model into a coherent logic model. Specifically, secondary cause development of the component malfunction and the basic event are coalesced into a new basic event. Command faults are not included in this coalescence[a].

The new basic event failure rate reflects the coalesced secondary causes as well as the old basic event causes of component malfunction. Also, an appropriate mean time to repair (MTTR) is assigned to the new basic event[b].

1. QUANTITATIVE PROCEDURES

Figure 1 is a diagrammatic respresentation, using fault tree symbology, of the general case of component malfunctions. Figure 2 shows an intermediate transformation of the logic model symbology given in Figure 1. The logic model in Figure 3 shows the final stage of transformation and is coherent. A failure rate and mean time to repair (MTTR) for B* must be determined. The intermediate stage of transformation is required if secondary faults, other than command faults, are present. The following notation is used for this determination:

$\theta(\ ) \equiv$ MTTR operator for the event appearing in the parentheses

$L(\ ) \equiv$ failure rate operator for the fault event appearing in the parentheses

$\tau_\ell \equiv$ MTTR of the ℓ^{th} event

---

[a] Actually, including the command faults in the coalescents is not improper; however, this inclusion is not necessary and can result in unnecessary dependency problems. Comments concerning these dependency problems will be given later.

[b] In this report, mean time to repair is used to describe the component repair characteristics. Component repair characteristics are described exactly by a time-dependent repair rate which implies a repair distribution. In practice, the mean time to repair is usually small when compared with the mean time to failure; in which case the exact repair distribution is of no practical consequence during system evaluations, but rather repair characteristics are adequately and conveniently described by the mean time to repair.

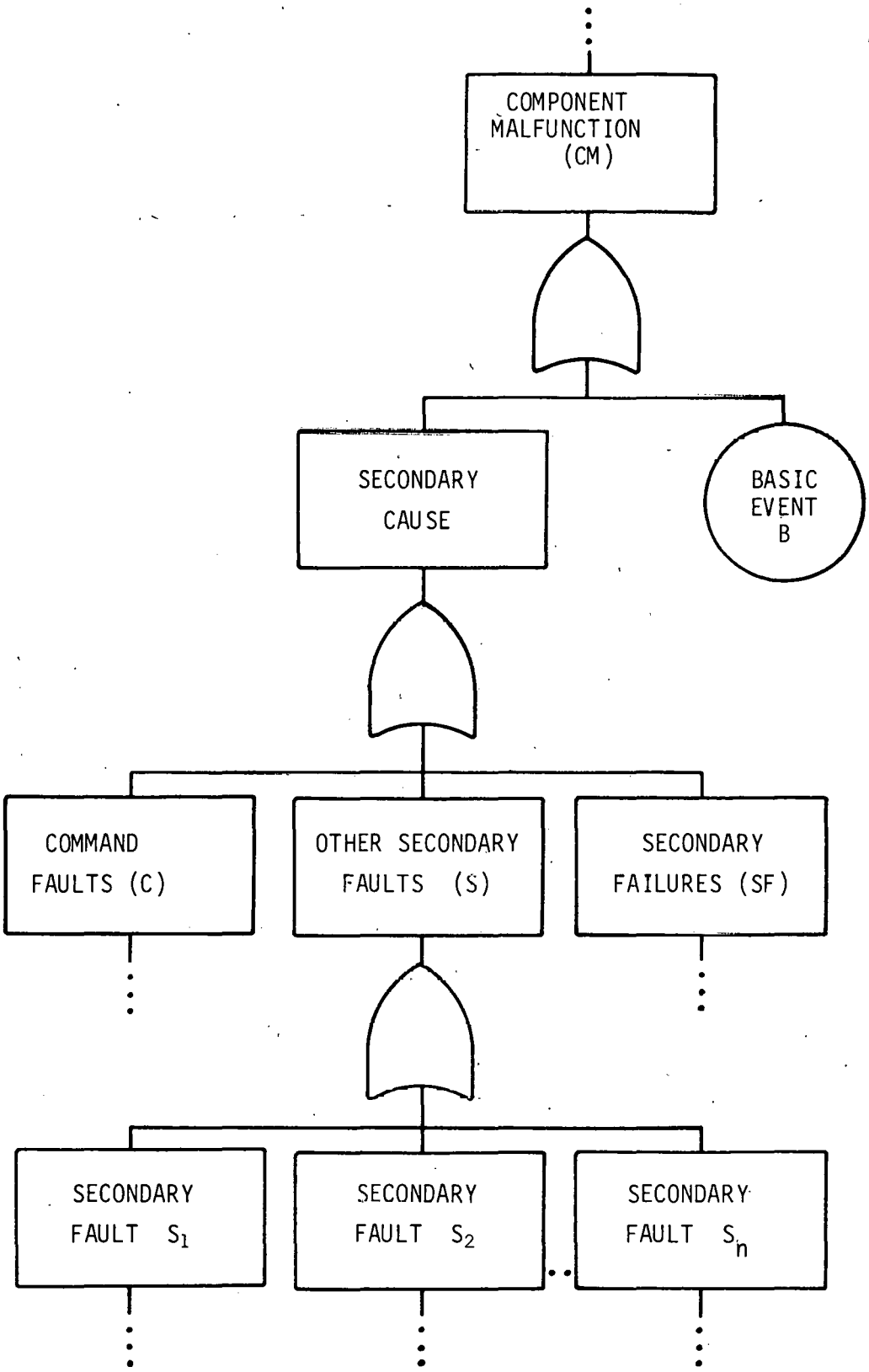


Fig. 1 General logic model representation of a component malfunction.

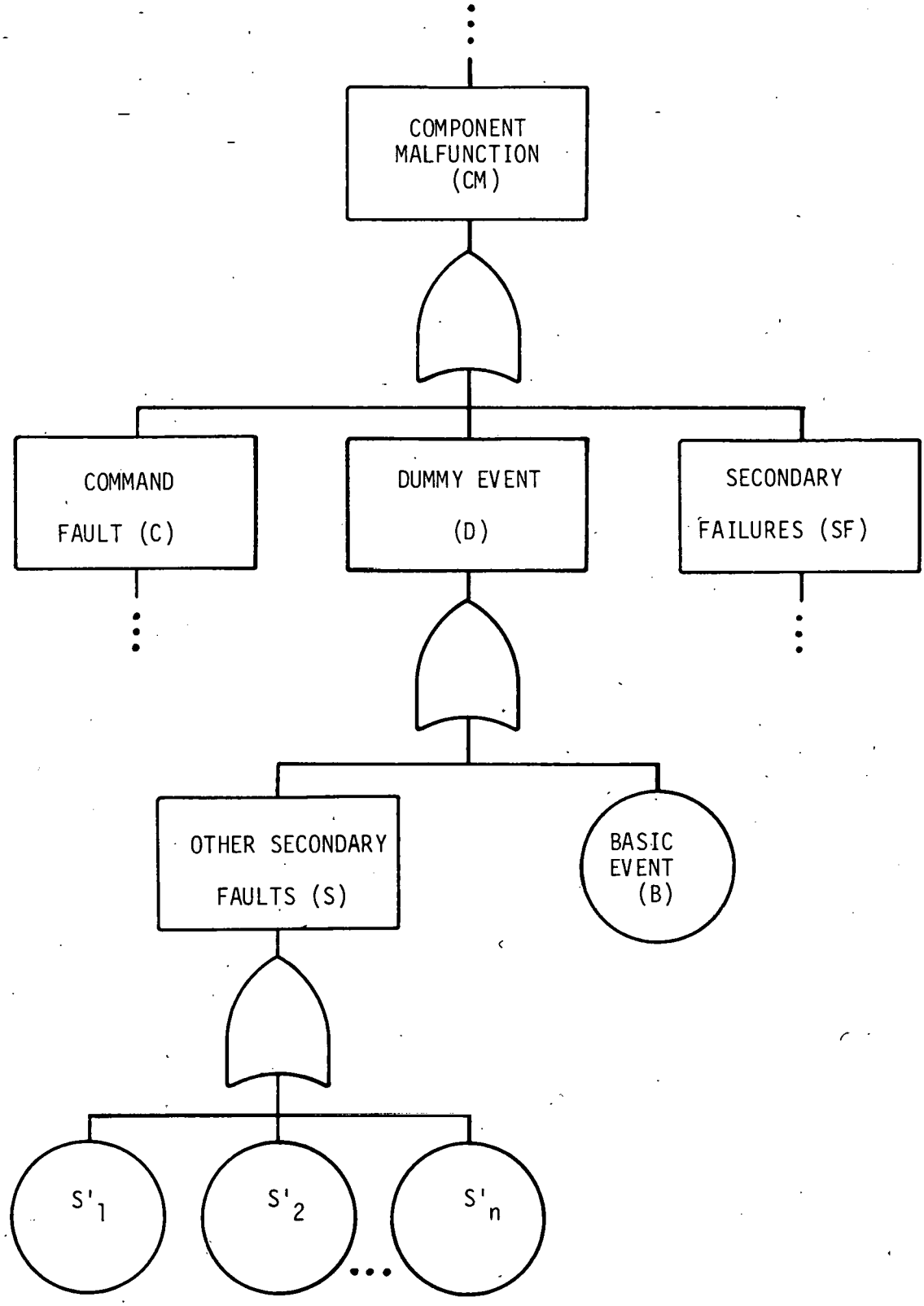


Fig. 2 Intermediate transformation step of the logic model shown in Figure 1.

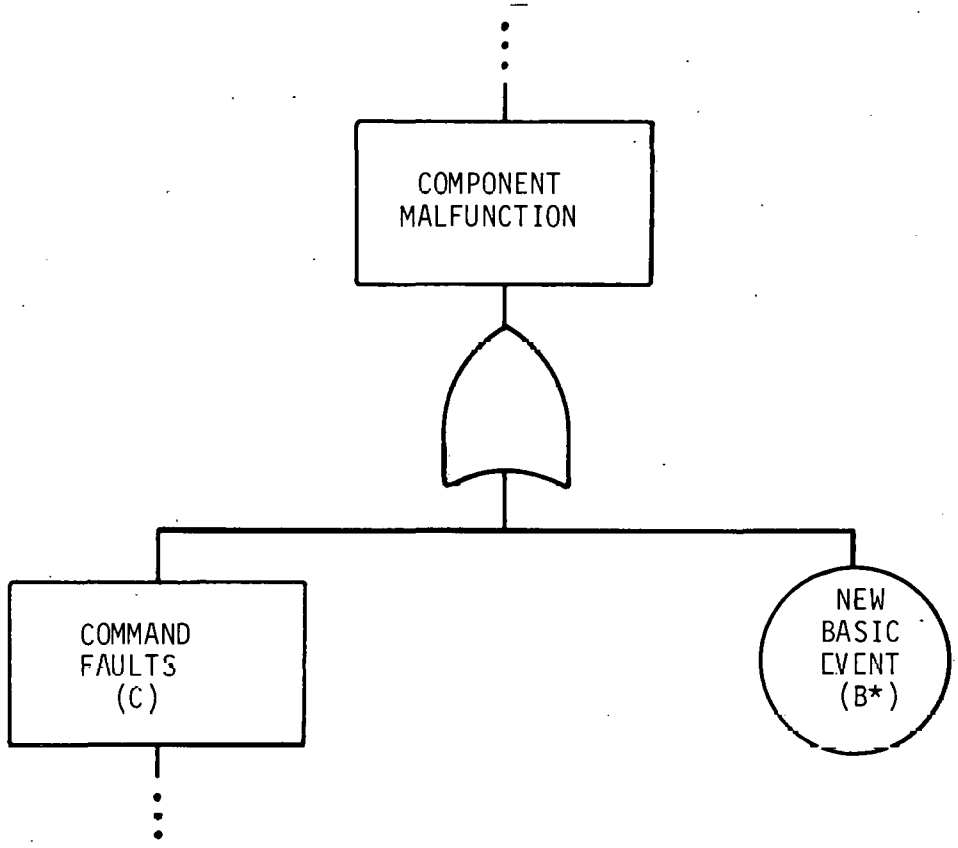


Fig. 3 Coherent transformation of the logic model shown in Figure 1.

$*$ ≡ superscript indicating a characteristic of a new event

τ_{h_i} ≡ mean time required for the component to again function properly, once the i^{th} secondary fault cause is removed

$'$ ≡ a superscript for events in the intermediate transformation

$''$ ≡ a superscript indicating the operation is performed from the intermediate transformation logic model

λ_i ≡ the failure rate of the i^{th} event.

In practice L( ) is a routine in the KITT computer program[1]. A method for θ( ) has been presented[9].

Procedures for obtaining characteristics for B* will be presented for two cases. The first case is appropriate if the component is repaired simultaneously with the secondary causes. The second case is used if the secondary causes are repaired simultaneously but the component is repaired only after all secondary causes have been repaired.

Case I: Procedure for Simultaneous Repair of Component and Secondary Failures

Steps 1 through 3 result in an evaluation of the characteristics of the basic events appearing in the intermediate transformation (Figure 2). Steps 4 through 7 result in the coalesced component malfunction in the coherent logic representation shown in Figure 3.

Step 1: Evaluate $\tau_{s_i} = \theta\ (S_i)$ for all i, i.e., all secondary faults

Step 2: Replace τ_{s_i} by $\tau_{s_i} + \tau_{h_i}$ for all i

Step 3: Evaluate $\lambda_{s_i} = L(S_i)$ for all i

Step 4: Evaluate $\tau_D = \theta''(D)$

Step 5: Evaluate $\tau_{SF} = \theta(SF)$

Step 6: Set $\tau_{B*} = \text{Max}\ (\tau_D, \tau_{SF})$

Step 7: Evaluate $\lambda_{B*} = L''(CM)$

Case II: Procedure for Simultaneous Repair of Secondary Failures Followed by Repair of Component

Same as Case I procedure except Step 6 is replaced by:

Step 6: Set $\tau_{B*} = \tau_D + \tau_{SF}$.

In practice these procedures are often simplified considerably because the general case is seldom experienced. For example, if only secondary failures are developed, Case I reduces to:

Step 1: Evaluate $\tau_{SF} = \theta(SF)$

Step 2: Set $\tau_{B*} = \text{Max}\ (\tau_B, \tau_{SF})$

Step 3: Evaluate $\lambda_{B*} = L(CM)$.

2. TREATMENT OF A FREQUENTLY ENCOUNTERED SPECIAL CASE

If a secondary failure has been developed using only OR logic gates and the MTTR of each basic event used in the development of the secondary failure is less than the MTTR of the basic event used for immediate development of the component malfunction (B in Figure 1), then no transformation is required if simultaneous repair of the component and its

secondary causes of failure is possible. All that is required is that the basic events used in the secondary failure development be assigned an MTTR equal to the MTTR of the basic event used for immediate development of the component malfunction.

This case is frequently encountered in practice. This simple treatment is also valuable because no dependency problem arises. These dependency problems will be commented on later.

V. SAMPLE PROBLEMS

Two examples of sample problems are given. The first is the application of the method to a liquid sodium pump to demonstrate the coalescing technique. Next, a sample system logic model is evaluated for system unavailability and unreliability.

1. EXAMPLE ONE - DEMONSTRATION OF COALESCING TECHNIQUE

The development of a sodium pump failing to provide flow in a sodium loop is shown in Figure 4. The basic event data are as follows:

Basic Event	Failure Rate λ(1/hr)	MTTR, τ (hr)
BE 1	10^{-6}	75
BE 2	10^{-4}	24
BE 3	10^{-3}	1

Repair of the sodium pump occurs only after the secondary causes have been removed (Case II).

The first step is to obtain an effective MTTR for all the secondary failures (SF). This MTTR, τ_{SF}, is given by

$$\tau_{SF} = \frac{\lambda_{BE2}\,\tau_{BE2} + \lambda_{BE3}\,\tau_{BE3}}{\lambda_{BE2} + \lambda_{BE3}} = 3.01 \text{ hr} .$$

Therefore, the MTTR of the coalesced event BE1* is given by

$$\tau_{BE1}{}^{*} = \tau_{BE1} + \tau_{SF} = 78 \text{ hr} .$$

The failure rate of BE1* is given by

$$\lambda^{*}_{BE1} = \lambda_{BE1} + \lambda_{BE2} + \lambda_{BE3}$$

$$= 1.101 \times 10^{-3} .$$

The final transformation is shown in Figure 5. No intermediate transformation is required because no secondary fault development is considered other than the command fault.

The unavailability of the sodium pump due to causes other than command faults, assuming erroneously that Figure 4 represents a coherent logic model, is given by

$$\bar{A} \approx \lambda_{BE1}\,\tau_{BE1} + \lambda_{BE2}\,\tau_{BE2} + \lambda_{BE3}\,\tau_{BE3}$$

$$\approx 0.0035 .$$

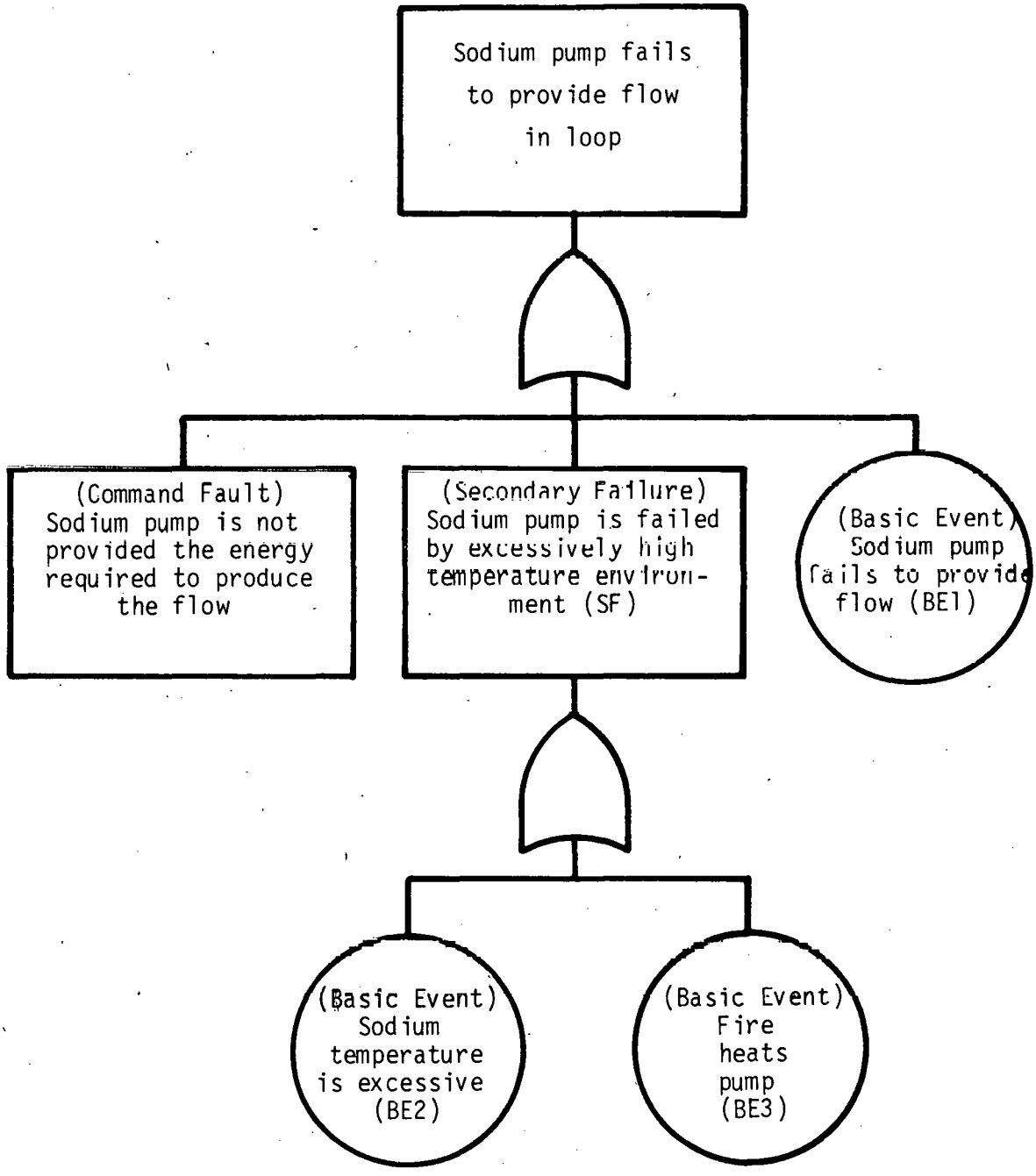


Fig. 4 Hypothetical development of a sodium pump malfunction.

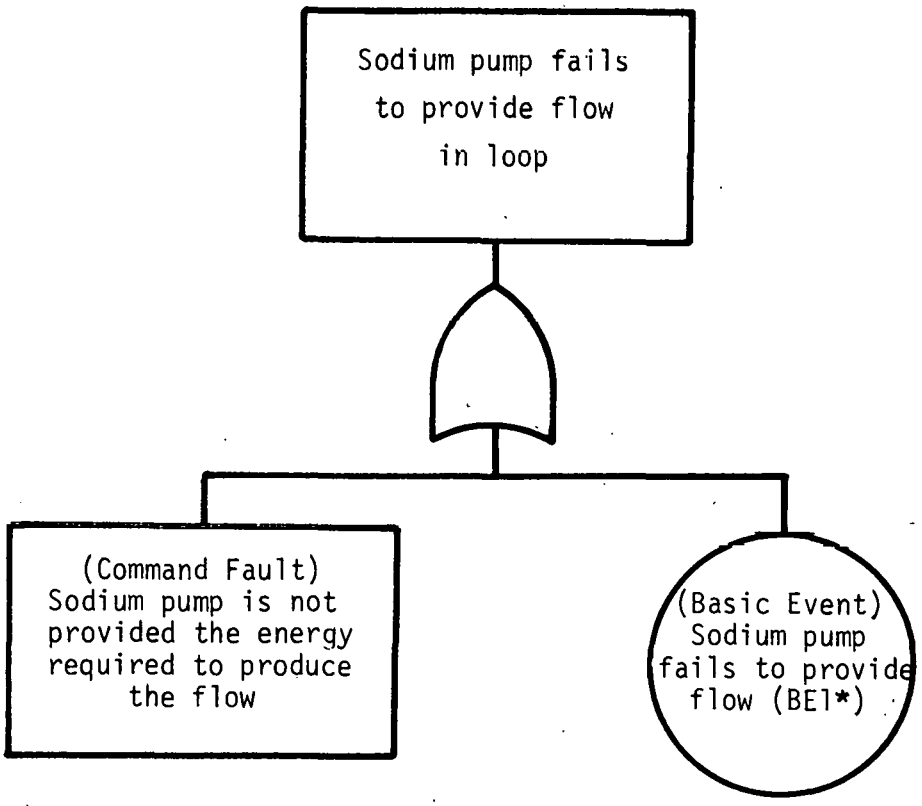


Fig. 5 Transformation of sodium pump development shown in Figure 4 into coherent logic.

The unavailability due to causes other than command faults, as calculated by the correct method presented here, is given by

$$\overline{A} \approx \lambda_{BE1}{}^{*} \cdot \tau_{BE1}{}^{*}$$

$$\approx 0.086 \ .$$

2. EXAMPLE TWO - EVALUATION OF SAMPLE SYSTEM LOGIC MODEL

For the next sample problem the logic model shown in Figure 6 is used. The intermediate transformation is not used because this transformation is useful for illustrative purposes only. The transformation into a coherent logic model is given in Figure 7.

The basic event input data are as follows:

Basic Event	Failure Rate, λ (1/hr)	MTTR, τ (hr)
B1	10^{-5}	100
B2	10^{-5}	10
B3	10^{-3}	5

Basic Event	Failure Rate, λ (1/hr)	MTTR, τ (hr)
B4	10^{-3}	10
B5	10^{-5}	20
B6	10^{-6}	50
B7	10^{-4}	10
B8	10^{-4}	1
B9	10^{-6}	50
B10	10^{-5}	100
B11	10^{-5}	1
B12	10^{-5}	1

Also $\tau_{h_{S1}} = 5$ hr

$\tau_{h_{S2}} = 10$ hr.

B2 and SF can be repaired simultaneously (Case 1).

To obtain the characteristics of B1*, the following procedure is used:

$$\lambda_{S1} \approx \lambda_{B5} + \lambda_{B3}\,\lambda_{B4}\,\tau_{B4} + \lambda_{B4}\,\lambda_{B3}\,\tau_{B3}$$

$$\approx 2.5 \times 10^{-5}$$

$$\tau_{S1} \approx \frac{\lambda_{B5}\,\tau_{B5} + \lambda_{B3}\,\tau_{B3}\,\lambda_{B4}\,\tau_{B4}}{\lambda_{S1}}$$

$$\approx 10$$

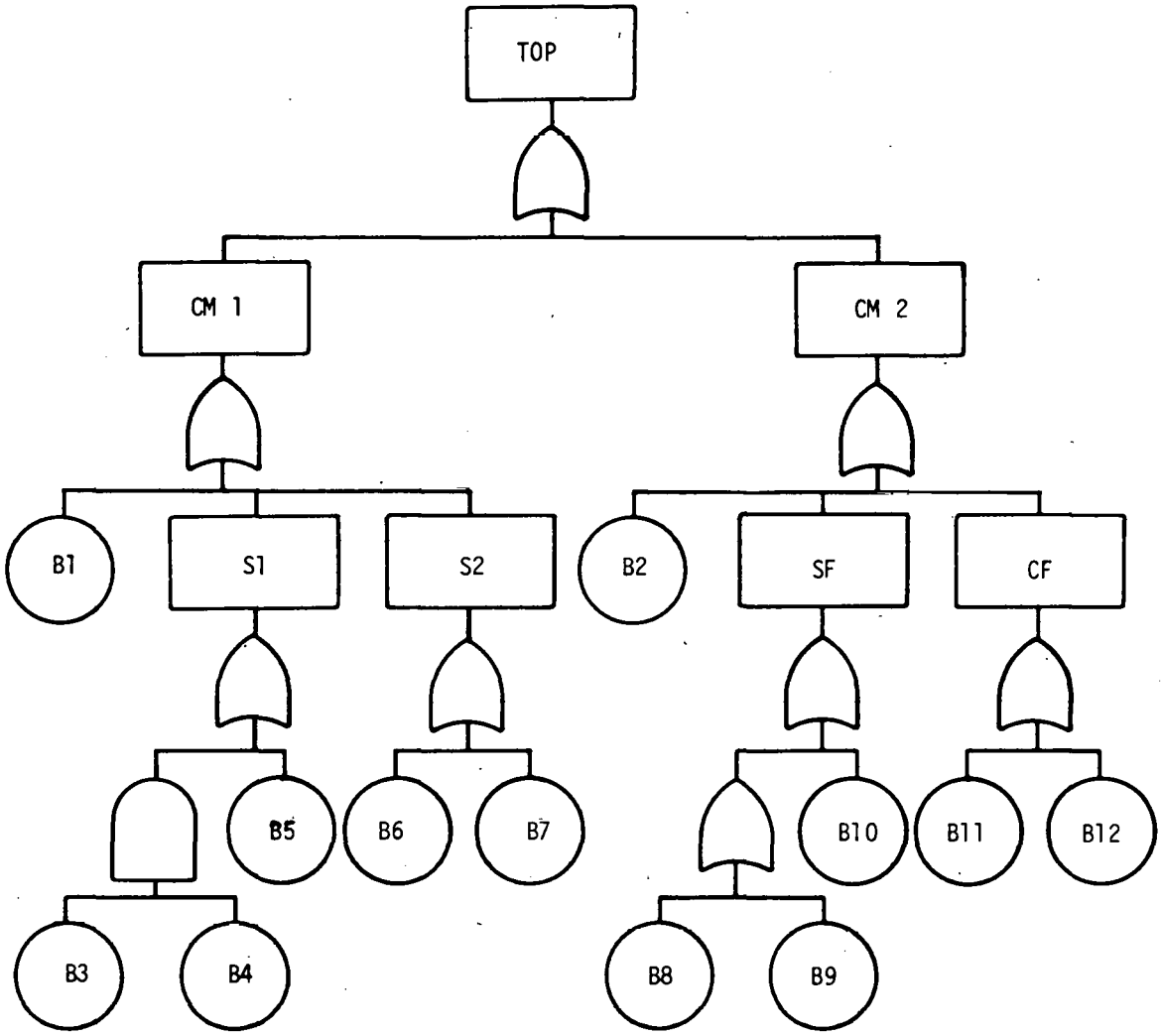
$$\tau^{*}_{S1} = \tau_{S1} + \tau_{h_{S1}} = 15 \text{ hr}$$

$$\lambda_{S2} \approx \lambda_{B6}\,\lambda_{B7}\,\tau_{B7} + \lambda_{B7}\,\lambda_{B6}\,\tau_{B6}$$

$$\approx 6 \times 10^{-9}$$

$$\tau_{S2} \approx \frac{\lambda_{B6}\,\tau_{B6}\,\lambda_{B7}\,\tau_{B7}}{\lambda_{S2}}$$

$$\approx 8.34 \text{ hr}$$



CM ≡ Component Malfunction

B ≡ Basic Event

CF ≡ Command Fault

S ≡ Secondary Fault other than CF

SF ≡ Secondary Failure

Fig. 6 Sample problem logic model.

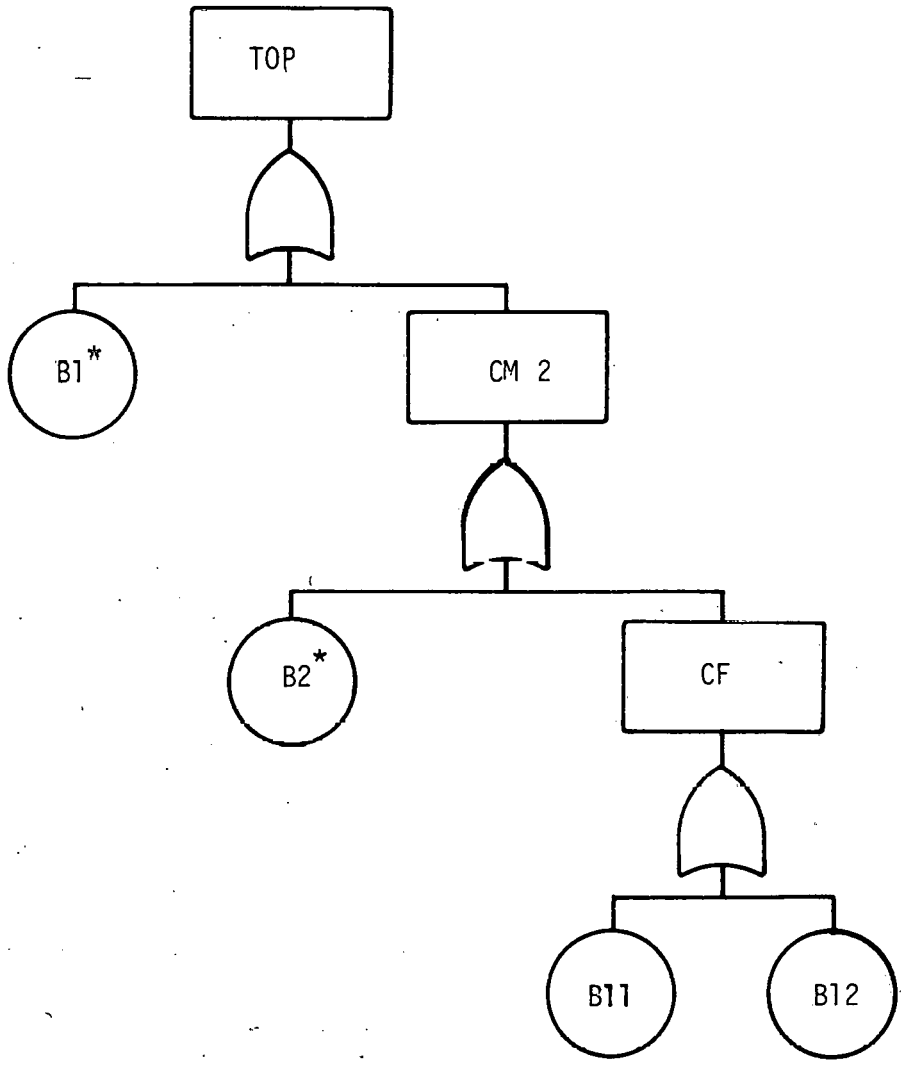


Fig. 7 Transformation of the logic model shown in Figure 6 into a coherent logic model.

$$\tau^*_{S2} = \tau_{S2} + \tau_{h_{S2}} = 18.34$$

$$\lambda_{B1}{}^* = \lambda_{B1} + \lambda_{S1} + \lambda_{S2}$$

$$\approx 3.5 \times 10^{-5}$$

$$\tau_{B1}{}^* \approx \frac{\lambda_{B1}\,\tau_{B1} + \lambda_{S1}\,\tau_{S1}{}^* + \lambda_{S2}\,\tau^*_{S2}}{\lambda_{B1}{}^*}$$

$$\approx 39.3 \text{ hr} .$$

To obtain the characteristics of B2* the following procedure is used:

$$\lambda_{SF} \approx \lambda_{B8}\,\lambda_{B10}\,\tau_{B10} + \lambda_{B10}\,\lambda_{B8}\,\tau_{B8} + \lambda_{B9}\,\lambda_{B10}\,\tau_{B10} + \lambda_{B10}\,\lambda_{B9}\,\tau_{B9}$$

$$\approx 1.025 \times 10^{-7}$$

$$\tau_{SF} \approx \frac{\lambda_{B8}\,\tau_{B8}\,\lambda_{B10}\,\tau_{B10} + \lambda_{B9}\,\tau_{B9}\,\lambda_{B10}\,\tau_{B10}}{\lambda_{SF}}$$

$$\approx 1.5 \text{ hr}$$

$$\tau_{B2}^* = \max(\tau_{SF}, \tau_{B2})$$

$$= 10 \text{ hr}$$

$$\lambda_{B2}^* \approx \lambda_{SF} + \lambda_{B2}$$

$$\approx 1.01 \times 10^{-5} \ .$$

The unavailability of the TOP event is given by

$$\bar{A}_{TOP} \approx \bar{a}_{B1}^*\,\bar{a}_{B1}^*\,\bar{a}_{B2}^* + \bar{a}_{B1}^*\,\bar{a}_{B11}^* + \bar{a}_{B1}^*\,\bar{a}_{B12}$$

where

$$\bar{a}_i = \lambda_i\,\tau_i$$

$$\approx 1.66 \times 10^{-7} \ .$$

The TOP event unreliability is given by

$$\bar{R}_{TOP} \approx \left(a_{B1}^*\,\lambda_{B2}^* + \bar{a}_{B2}^*\,\lambda_{B11}^* + \bar{a}_{B1}^*\,\lambda_{B11} + \bar{a}_{B11}\,\lambda_{B1}^* + \bar{a}_{B1}^*\,\lambda_{B12} + \bar{a}_{B12}\,\lambda_{B1}^*\right) t$$

$$\approx \left(4.56 \times 10^{-8}\right) t \ .$$

VI. COMMENTS ON DEPENDENCIES

After the transformation, the logic model is coherent; however, all the basic events in the final logic model are not necessarily independent even though the basic events in the original logic model were independent. This dependence results from common events in the development of the secondary causes among the various component malfunctions appearing in the logic model. Treatment of this dependency is beyond the capability of any presently available technique. The dependency is, however, explicitly defined by the logical development that is coalesced. Therefore, exact quantitative treatment of the dependency is feasible.

VII. CONCLUSION

Consideration of the details of component malfunction causes is necessary for both qualitative and quantitative system reliability and safety analysis. This report has put forth the framework on which future method development efforts can be based.

The case of no duplicate basic events among the coalesced secondary causes has been covered in this report. Duplication of component malfunction development is handled correctly using the procedure presented.

The case of partial repetition of logical development among the coalesced secondary causes cannot be handled by using the techniques presented here. The procedure required for treatment of TOP event unavailability in this case is straightforward but beyond the scope of this report. Preliminary studies indicate that the treatment of TOP event unreliability and expected numbers of failures is feasible although much more difficult.

Computer aid for analysis of system logic models containing secondary cause development must be developed. The analysis in practice is too complex to be carried out by manual procedures.

The consideration of details during the analysis of component malfunction is far from academic since, if these details are not considered, the decisions based on qualitative or quantitative analysis can be in error.

VIII. REFERENCES

1. W. E. Vesely, "A Time-Dependent Methodology for Fault Tree Evaluation", *Nuclear Engineering and Design, 13* (August 1970) 337-360.

2. R. E. Barlow and P. Chatterjee, *Introduction to Fault Tree Analysis,* ORC 73-30 (December 1973), p 24[a].

3. W. E. Veseley, *Analysis of Fault Trees by Kinetic Tree Theory,* IN-1330 (October 1969)[b].

4. P. A. Crosetti, *Computer Program for Fault Tree Analysis,* DUN-5508(April 1969)[b].

5. H. E. Kongso, *RELY4: A Monte Carlo Computer Program for System Reliability Analysis,* RISØ-M-1500 (June 1970).

6. J. D. Esary and F. Proschan, "A Reliability Bound for Systems on Maintained, Interpendent Components", *Journal of the American Statistical Association, 65,* 329 (March 1970) pp 329-338.

7. S. M. Ross. *Multicomponent Reliability Systems,* University of California, Berkeley, ORC 74-4 (February 1974)[a].

8. P. M. Nagel, "A Monte Carlo Method to Compute Fault Tree Probabilities", *System Safety Symposium, June 8-9, 1965, Seattle, The Boeing Company.*

9. J. B. Fussell, "How to Hand-Calculate System Reliability and Safety Characteristics", *IEEE Transactions on Reliability, R-24,* 3 (August 1975).

10. D. F. Haasl, "Advanced Concepts in Fault Tree Analysis", *System Safety Symposium, June 8-9, 1965, Seattle, The Boeing Company,* Available from the University of Washington Library, Seattle, Washington.

11. R. E. Barlow and F. Proschan, *Statistical Theory of Reliability and Life Testing,* New York: Holt, Rinehart, and Winston, Inc., 1975.

---

[a] Available from the NTIS, Springfield, VA, 22151, USA.

[b] Available from the Operations Research Center, University of California, Berkeley, Ca 94720

APPENDIX A

APPROXIMATION METHODS

APPENDIX A

APPROXIMATION METHODS

This appendix contains methods, used in this report, for calculating unreliabilities and unavailabilities for basic events, cut sets, and fault tree TOP events.

1. NOTATION

s-	prefix, implies "statistical"
f_k	probability density function of time to the first failure of minimal cut set k
MTF_t	mean time to failure of the TOP event
$\overline{a}_i, \overline{A}_k, \overline{A}_t$	unavailability of basic event i, of minimal cut set k, and of the TOP event
i	basic event i in the minimal cut set k
n_k	number of basic events in minimal cut set k
N	number of crucial minimal cut sets
$\overline{r}_i, \overline{R}_k, \overline{R}_t$	unreliability of basic event i, of minimal cut set k, and of the TOP event
$\lambda_i, \Lambda_k, \Lambda_t$	failure rate of basic event i, of minimal cut set k, and of the TOP event
τ_i, τ_t	mean dead time of basic event i, and of the TOP event
t	time

2. BASIC EVENT FAILURE INFORMATION

For basic events, repairable or not, with constant failure rates[A-1]:

$$\overline{r}_i \lesssim \lambda_i t \quad . \tag{A-1}$$

For nonrepairable components:

$$\bar{a}_i = \bar{r}_i \quad . \tag{A-2}$$

For repairable components[A-2]:

$$\bar{a}_i = \left[\lambda_i\tau_i/(1 + \lambda_i\tau_i)\right] [1 - \exp-(\lambda_i + 1/\tau_i)t] \tag{A-3}$$

$$\bar{a}_i \lesssim (\lambda_i\tau_i)/(1 + \lambda_i\tau_i) \tag{A-4}$$

$$\bar{a}_i \lesssim \lambda_i\tau_i \quad . \tag{A-5}$$

Approximations (A-1), (A-4), and (A-5) overpredict $\bar{r}_i$ and $\bar{a}_i$ at all times. When $\lambda_i t < 0.1$, the overprediction by Approximation (A-1) is small. At times greater than $2\tau_i$ and $3\tau_i$, Approximation (A-4) overpredicts no more than 5% and 14%, respectively. If $\lambda_i\tau_i << 0.1$, then Approximation (A-5) can be used instead of Approximation (A-4).

3. MINIMAL CUT SET FAILURE INFORMATION

Since all basic events must be independent and all the basic events in a minimal cut set must exist for the minimal cut set failure to occur:

$$\bar{A}_k = \prod_{i=1}^{n_k} \bar{a}_i \quad . \tag{A-6}$$

For minimal cut sets with all nonrepairable basic events, the unreliability is identical in value to the unavailability. Therefore, for nonrepairable minimal cut sets, Equation (A-6) can be used for the unreliability.

In general, for systems for which some or all of the basic events are repairable, by definition

$$f_k dt = \Pr\{B \cap C\} \Pr\{D \mid B \cap C\}$$

where, for minimal cut set k,

B = the event the failure exists at time t + dt

C = the event the failure does not exist at time t

D = the event the failure has not occurred to time t.

For reliable minimal cut sets, $P(D \mid C \cap B)$ is close to unity. In any case, it is less than or equal to unity; therefore:

$$f_k dt \leq \Pr\{B \cap C\} \quad .$$

The event $B \cap C$ can occur in n_k mutually exclusive ways. The first basic event in the minimal cut set can occur in the time interval t to t + dt with the remaining events having already occurred at time t, or the second basic event can occur in the time interval t to t + dt with the remaining events already having occurred or. . .

Therefore, as has been shown by Veseley[A-3]:

$$\Pr\{B \cap C\} = \sum_{j=1}^{n_k} a_j \lambda_j dt \prod_{\substack{i=1 \\ i \neq j}}^{n_k} \bar{a}_i \quad . \tag{A-7}$$

In most cases of interest, a_j is near unity. In any case, the inequality is preserved by setting "$a_j = 1$". In which case

$$f_k \leq \bar{A}_k \sum_{j=1}^{n_k} \lambda_j / \bar{a}_j \quad . \tag{A-8}$$

The unreliability of the minimal cut set, $\bar{R}_k$, is

$$\bar{R}_k = \int_0^t f_k \, dt \quad . \tag{A-9}$$

Equation (A-9) is easily evaluated when Equation (A-8) is used for f_k, because the integrand is a polynomial in t when Equations (A-1), (A-3), or (A-5) are used to evaluate the a_j; Equation (A-9) is only a slight overprediction of the unreliability when $\bar{R}_k < 0.1$.

By definition, the minimal cut set failure rate is

$$\Lambda_k = f_k / R_k \quad . \tag{A-10}$$

4. TOP EVENT FAILURE INFORMATION

The occurrence of any one minimal cut set failure will cause the TOP event to occur; therefore:

$$\bar{A}_t \leq \sum_{k=1}^{N} \bar{A}_k \quad . \tag{A-11}$$

A somewhat better overpredicting approximation for $\bar{A}_t$, but which is more tedious to evaluate, is[A-4]

$$\bar{A}_t \leq 1 - \prod_{k=1}^{N} A_k \quad .$$

In the nonrepairable case the result of (A-11) can be used for the system unreliability.

As shown in Equation (A-5) the TOP event unreliability is, in general, bounded as follows:

$$\bar{R}_t < 1 - \prod_{k=1}^{N} R_k \quad . \tag{A-12}$$

Therefore, since $R_t < 1$, the system unreliability is overpredicted by

$$\bar{R}_t \leq \sum_{k=1}^{N} \bar{R}_k \quad . \tag{A-13}$$

If the system reliability is closely approximated by Equation (A-12), the usual practical case, then the TOP event failure rate is

$$\Lambda_t \simeq \sum_{k=1}^{N} \Lambda_k \quad . \tag{A-14}$$

Overprediction at all times of Λ_t for repairable systems by Equation (A-14) has not yet been proved.

If all the basic events are repairable, or if all the nonrepairable basic events are in one event minimal cut sets, Λ_t approaches a constant[a], and the mean time to failure of the TOP event is

$$MTF_t \simeq 1/\Lambda_t \quad . \tag{A-15}$$

Since $\bar{A}_t/A_t$ is the ratio of the expected system downtime to the expected system uptime, the expected time the system will be failed, given it has failed or, equivalently, the TOP event mean dead time, is

$$\tau_t \simeq \bar{A}_t/(A_t\Lambda_t) \quad . \tag{A-16}$$

A more rigorous treatment of MTF_t and τ_t for repairable systems has been presented by Ross[A-6].

[a] Use of the approximation herein for estimating Λ_t results in Λ_t being a weak monotonic function of time even if all primary events are repairable. Therefore, Λ_t should be estimated at the maximum system mission time. Actually Λ_t asymptotically approaches a constant in the totally repairable case.

5. REFERENCES

A-1. N. H. Roberts, *Mathematical Methods in Reliability Engineering,* New York: McGraw-Hill Book Company, Inc., 1964, p 152.

A-2. M. L. Shooman, *Probablistic Reliability: An Engineering Approach,* New York: McGraw-Hill Book Company, Inc., 1968.

A-3. W. E. Vesely, "A Time-Dependent Methodology for Fault Tree Evaluation", *Nuclear Engineering and Design, 13* (August 1970) pp 337-360.

A-4. R. E. Barlow and P. Chatterjee, *Introduction to Fault Tree Analysis,* University of California at Berkeley, ORC 73-30 (December 1973)[a] p 24.

A-5. J. D. Esary and F. Proschan, "A Reliability Bound for Systems of Maintained, Interdependent Components", *Journal of the American Statistical Association, 65,* 329 (March 1970) pp 329-338.

A-6. S. M. Ross, *Multicomponent Reliability Systems,* University of California at Berkeley, ORC 74-4 (February 1974)[a].

[a] Available from the Operations Research Center, University of California, Berkeley, 94720, USA.

REPORT II

TECHNIQUES FOR QUALITATIVE ANALYSIS OF COMMON CAUSE FAILURES

J. R. Wilson
J. B. Fussell
G. R. Burdick
D. M. Rasmuson
J. C. Zipperer

ABSTRACT

A workable qualitative analysis technique is presented to locate common causes of system failure. New concepts are introduced that allow computer programs to be used as an aid in the analysis. The report is written for engineers with basic training in reliability and safety engineering techniques.

ACKNOWLEDGMENTS

We gratefully acknowledge the contributions of M. E. Stewart, R. J. Crump, and J. E. Trainer during the formative stages of this effort. Also, D. F. Haasl, Institute of Systems Sciences, is acknowledged for his contributions.

CONTENTS FOR REPORT II

FIGURES

TABLES

TECHNIQUES FOR QUALITATIVE ANALYSIS OF COMMON CAUSE FAILURES

I. INTRODUCTION AND DEFINITIONS

Analyzing common cause events is a part of system reliability and safety analysis. A common cause event, often called a common mode failure, is a secondary cause[a] that is applicable to the development of more than one component malfunction. Although common cause events have been of considerable concern in practice, only a small portion of the literature has been devoted to this subject; the reason being that without a well-defined structure, study of common cause events is not generally tractable.

The importance of considering common cause events was reported by Epler[1] in 1961, as follows:

> "This raises serious doubts as to the usefulness of a reliability calculation that considers random events only, when the common mode failure may be dominant by as much as a factor of 10^5. However, a concentrated attack on this problem by both designers and operators might improve the common mode failure rate from 10^{-2} to 10^{-3} per year. Even after such an improvement, the common mode failure would remain dominant. This position, based on ORNL experience, is in substantial agreement with Laurence. [George C. Laurence, Reactor Safety in Canada, Nucleonics, 18 (10), 73077 (October 1960).]"

Common cause events are not universally considered to be dominant events, however. In the Reactor Safety Study[2] the following statement is made concerning common cause events other than human error:

> "Common mode failures [excluding human causes] in many cases did not have significant effects. Single system failure probabilities dominated the accident sequence probability, and single component failures, in turn, dominated the system probability. When this occurred, common mode failures thus had little impact since at most they could change multiple failures into single failures and these [failures of this order] already existed."

On the other hand, in the same study another source of common cause events, human errors[b], "in a number of cases dominated the system, because of their larger basic probabilities as compared to component failure rate data"[2].

---

[a] Discussed in Report I of this document.

[b] Human errors were not considered as common cause events in Reference 2.

At this point, definition of several terms is necessary. A significant common cause event is a secondary cause that is common to all the basic events in one or more minimal cut sets[3]. The minimal cut set for which the significant cause event is applicable is called a common cause candidate. In addition, if all the components represented by the basic events in that minimal cut set share a "common location", that minimal cut set is a prime common cause candidate. Components share a common location if no barriers are present that are capable of insulating the components from the secondary cause. Components may share a common location irrespective of the physical distance separating them.

By limiting a study to system cut sets the analysis for common causes becomes tractable because

(1) No additional basic events need to be added to the logic model

(2) No additional minimal cut sets result

(3) Analysis for common causes becomes an option that can be exercised, without foreplanning, after other types of analyses are complete

(4) Computer aided analysis can be used advantageously.

The methods presented in this report are concerned with locating common cause candidates and prime common cause candidates by identifying the associated significant common cause events.

On occasion, a significant common cause event may not be specified for a prime common cause candidate, but rather the prime common cause candidate is identified soley on the basis of a "common link condition". A common link is a condition that closely links all the basic events in the minimal cut set. The probability that the condition exists at the time of analysis is assumed to be unity. For example, all components indicated by the basic events in a minimal cut set being produced by the same manufacturer is a common link. The prime common cause candidate is then identified without concern about a common location. Other common link conditions arise from components being tightly linked by a common location. Components in the same electrical circuit, chemical flow loop, or even tightly clustered in a cabinet can give rise to prime common cause candidates based on common link conditions rather than as the result of specifying the secondary cause susceptibility and location of each component.

The purpose of this report is not to suggest methods of quantitative evaluation concerning these prime common cause candidates, but rather to provide techniques for detailed qualitative analysis. Because of the substantial amount of information that must be considered during analysis of situations encountered in practice, the methodology presented is formulated specifically for computer-aided analysis.

Section II of this report introduces the subject of generic classification and tabulation of secondary causes of component malfunctions and conditions that can result in prime common cause candidates. Section III presents the proposed input format for the computer program. Section IV gives additional information concerning representation of the secondary cause susceptibilities for the individual components. Section V gives the method for entering the common locations into computer coding sheets. Finally conclusions and recommendations are made in Section VI.

II. COMMON LINKS AND GENERIC CAUSE LIST

A tremendous number of secondary failure causes are possible. As a result the analysis is subject to omissions or redundancies (representing the same failure mode by different sources, for example, including both "water hammer" and "pipe whip"). Redundancy can largely be eliminated by listing only generic causes (each cause represents a class of conditions or secondary failure causes). Omissions can be minimized by organizing the generic causes into natural groupings, or categories, which aids in the selection of entries for the list; the basis for the formation of these categories is the nature of the generic cause. In addition, breaking up the list into these categories not only helps the analyst by reducing his field of consideration, but it greatly simplifies the computer search techniques to be developed at a later date. The purpose here is not to break the causes down so finely that physical meaning is lost, but rather to eliminate redundancy (combining "fire" and "high temperature" or "flood" and "moisture").

The computer aid to be developed requires the analyst to consider only the most significant generic causes in each of four broad categories (mechanical or thermal, electrical or radiation, chemical or miscellaneous, and common links) for each failure event. The generic causes in these categories are given in Tables I through IV. A suggested generic list by category -- which can be easily updated without methodology modification – is given in Sections II-1 through II-4.

TABLE I

GENERIC CAUSES OF A MECHANICAL OR THERMAL NATURE

Symbol	Generic Cause	Example Sources
I	Impact	Pipe whip, water hammer, missiles, earthquake, structural failure
V	Vibration	Machinery in motion, earthquake
P	Pressure	Explosion, out-of-tolerance system changes (pump overspeed, flow blockage)
G	Grit	Airborne dust, metal fragments generated by moving parts with inadequate tolerances
S	Stress	Thermal stress at welds of dissimilar metals, thermal stresses and bending moments caused by high conductivity and density of liquid sodium
T	Temperature	Fire, lightning, welding equipment, coolant system faults, electrical short circuits

TABLE II

GENERIC CAUSES OF AN ELECTRICAL OR RADIATION NATURE

Symbol	Generic Cause	Example Sources
E	Electromagnetic Interference (EMI)	Welding equipment, rotating electrical machinery, lightning, power supplies, transmission lines
R	Radiation damage	Neutron sources, sources of ionizing radiation
M	Conducting Medium	Moisture and conductive gases
V	Out-of-tolerance voltage	Power surge
I	Out-of-tolerance Current	Short circuit

TABLE III

GENERIC CAUSES OF A CHEMICAL OR MISCELLANEOUS NATURE[a]

Symbol	Generic Cause	Sample Sources
A	Corrosion (acid)	Boric acid from neutron control systems; acid used in maintenance for removing rust and cleaning
O	Corrosion (oxidation)	Water medium, high temperature metals (filaments)
R	Other chemical reactions	Galvanic corrosion; the complex interactions of fuel cladding, water, oxide fuel, and fission products; leaching of carbon from stainless steel by sodium
C	Carbonization	Oil in liquid sodium
B	Biological hazards	Poisonous gases, explosions, missiles

[a] Sodium-water and sodium-air reactions have been left out of the table because the resulting failure modes can be represented by other generic causes: temperature and biological hazards. However, the analyst, for clarity, may expand the table to include sodium reactions.

TABLE IV

COMMON LINKS RESULTING IN DEPENDENCE BETWEEN COMPONENTS

Symbol	Common Link	Example of situation that can result in system failure when all basic events in a minimal cut set share the same common link.
E	Energy Source	Common drive shaft, same power supply
C	Calibration	Misprinted calibration instructions
F	Manufacturer	Repeated fabrication error, such as neglecting to properly coat relay contacts
I	Installation contractor	Same subcontractor or crew
M	Maintenance	Incorrect procedure, inadequately trained person
O	Operator or operation	Operator disabled or overstressed; faulty operating procedures
P	Proximity	Location of all components of a cut set in one cabinet. This exposes all of them to many unspecified common causes
T	Test procedure	Faulty test procedures which may affect all components normally tested together
N	Energy flow paths	Location in same hydraulic loop, location in same electrical circuit
S	Similar parts	Important in the case of minimal cut sets which contains only pumps, or only valves, etc.

Through use of tables of generic causes (for example, Tables I through IV), the analyst chooses those causes applicable to this analysis, adds quantifying details (for example, temperature over 800°F) and combines causes, where desired (for example, conducting medium, oxidation, and high temperature represent steam; or impact and vibration to represent earthquake).

The table heading represents the nature of the generic causes which follow. In the discussions on each table, certain generic causes are elaborated upon for the sake of clarity.

1. MECHANICAL OR THERMAL GENERIC CAUSES

Impact can be differentiated from vibration on the basis of the duration of force. An impact is an application of force over small time interval such as the blow from a pipe (pipe whip) or the effect from a flying particle or missile. A vibration is an oscillating force, destructive due to its persistance, oscillation, and amplitude of the force.

2. ELECTRICAL OR RADIATION GENERIC CAUSES

Electromagnetic interference (EMI) comes from many sources that include welding equipment, rotating electrical machinery, diodes, transistors, transmission lines, neon and florescent lights, power supplies, and lightning. For many of these, systems are protected by design features, such as distance, shielding, and coaxial cables. EMI causes which are of sufficient magnitude and are not eliminated by design features become considerations in the common cause analysis. Table II gives a listing of topics of consideration for electrical or radiation generic causes.

A conducting medium (for example, a gas such as the argon cover gas in a sodium system, which has exceeded its breakdown voltage) could cause shorting, arcing, and other ionization effects if it is present in sufficient quantities around electrical equipment. Such electrical effects can occur due to breakdown of ambient gases under normal high voltage conditions.

3. CHEMICAL OR MISCELLANEOUS GENERIC CAUSES

Some of the secondary causes listed in Table III are not strictly generic (like carburization, a problem in sodium systems), but were considered important enough to be listed separately. "Other chemical reactions" is a complex secondary cause. This cause concerns reactions which, given certain conditions, will cause extensive interactions. For instance, impurities in a sodium system can cause leaching throughout the system in addition to the reactions due to pure sodium. A similar impurity problem in water reactors is fission gas (I_{131}) from failed fuel pins causing decarburization in stainless steel components.

An example of a subtle chemical common cause problem is "residual binder". Sometimes the manufacturing process does not remove all the carbon binder used to construct the fuel pellets. In the reactor this excess carbon forms CO_2, creating unexpected additional gas pressure. This example is one of a local reaction, limited to the internals of a fuel pin, which can be repeated simultaneously throughout the core (in all fuel pins manufactured with "residual binder").

Oxidation corrosion may be combined with stress (as discussed in Section II-1, Mechanical or Thermal Generic Causes) to represent the stress-corrosion secondary cause. Biological hazards (poisonous gases, explosions, flying missiles) may disable all or a part of the operating crew, depending upon the area encompassed (as discussed in Section IV, Details on Constructing Domains).

"Miscellaneous" has been added to the category heading to allow for analysis of additional generic causes which do not properly fit elsewhere.

1. COMMON LINKS

The common links category allows the analyst to account for common links among components in a system that increase the probability of a number of components failing. These common linking conditions are conceptually different from the generic cause susceptibility categories.

No significant common cause event is given for the prime common cause candidate identified on the basis of these common links. For example, if all the components indicated by a minimal cut set are linked by the same electrical circuit, the resulting dependence creates a common cause candidate that is based on this situation alone with no significant common cause event specified. Consequently, there is no checking of location for any common cause candidate based on this category.

Detailed treatment of these common links is key to a meaningful common cause analysis. The treatment often requires that attention be given to subtle aspects of the system. For example two subsystems may appear safe because they are separated, but may share parallel functions. These parallel functions may cause the subsystems to be subject to the same secondary causes. Specifically, the coolant loops on a reactor may be located physically apart but may share the same test, maintenance, and operation procedures (Table IV). A maintenance man (using the wrong oil, for instance) working on both subsystems may circumvent the design redundancy.

The following calibration error having potential widespread effects occurred at the Oak Ridge Research Reactor[1].

> "In an effort to improve maintenance procedures, instrument settings were typed and pasted near the related instruments. It was discovered that the typist had made an error and all identical instruments would have been incorrectly set."

An example of the proper use of the maintenance common link would be to include only those failure events for which the failure probability is significantly increased by faulty maintenance or lack of maintenance. Passive elements (pipes, vessels), for instance, are not greatly affected by maintenance, but some active elements are affected (instrumentation and controls).

The following incident which could be attributed to an operation common link occurred at the Materials Testing Reactor[1]:

> "It was so arranged that each aluminum rabbit tube could, on occasion, be withdrawn into the body of a gate valve that also could be used to cut off water flow through the tube. On this occasion all the rabbit tubes were within the valves, and, to prevent the valves being closed and thereby destroying the tubes, all valve handles had been removed.
>
> A young engineer working for the operating contractor came in to close the valves. A painter working for the construction contractor informed the engineer that the valve handles had been removed by the painter's supervisor because for some reason the supervisor did not want them closed. The young engineer said that he had been instructed to close the valves, which he then did with a pipe wrench; this destroyed all the rabbit tubes."

III. COMPUTER REPRESENTATION OF BASIC EVENTS AND GENERIC CAUSE SUCEPTIBILITIES

The format to be used for the computer program for common cause analysis is compatible with the input format used with computer programs for qualitative and quantitative reliability and safety analysis such as PREP[4], KITT[4], and MOCUS[5]. This proposed format is shown in Figure 1.

1. BASIC EVENT IDENTIFICATION

An eight-character computer word represents the basic event. This basic event identification involves a system code (such as electrical power, reactor, or reactor protection), component type code (air-operated valve, diesel, or pipe), component identifier (to render each component distinct), and fault mode code (does not close, rupture, short,...). For example, the event name, LAMA108Q, would be interpreted as follows:

L	AM	A108	Q
Electrical Power	Amplifier	Amplifier 08 on Chart A1	Short to Power

The system code, component type code, and fault mode code are explained in Appendix A. The component identifier code is specified by the analyst. The analyst may employ the basic event identification of his choice, but the preceding code was selected because it is convenient and consistent with the Reactor Safety Study[2].

2. COMPONENT RELIABILITY CHARACTERISTICS

The failure rate (λ) is contained in Columns 11 to 20 in figure 1, and the repair rate (τ) is contained in Columns 21 to 30. For the option of Monte Carlo runs as described in the reactor safety study, the error factors (parameter distribution) for λ and τ appear in Columns 32 to 36 and 38 to 42.

3. SUBTREE IDENTIFICATION

The subtree identification flag is used when a large fault tree is broken into several subtrees to be analyzed separately.

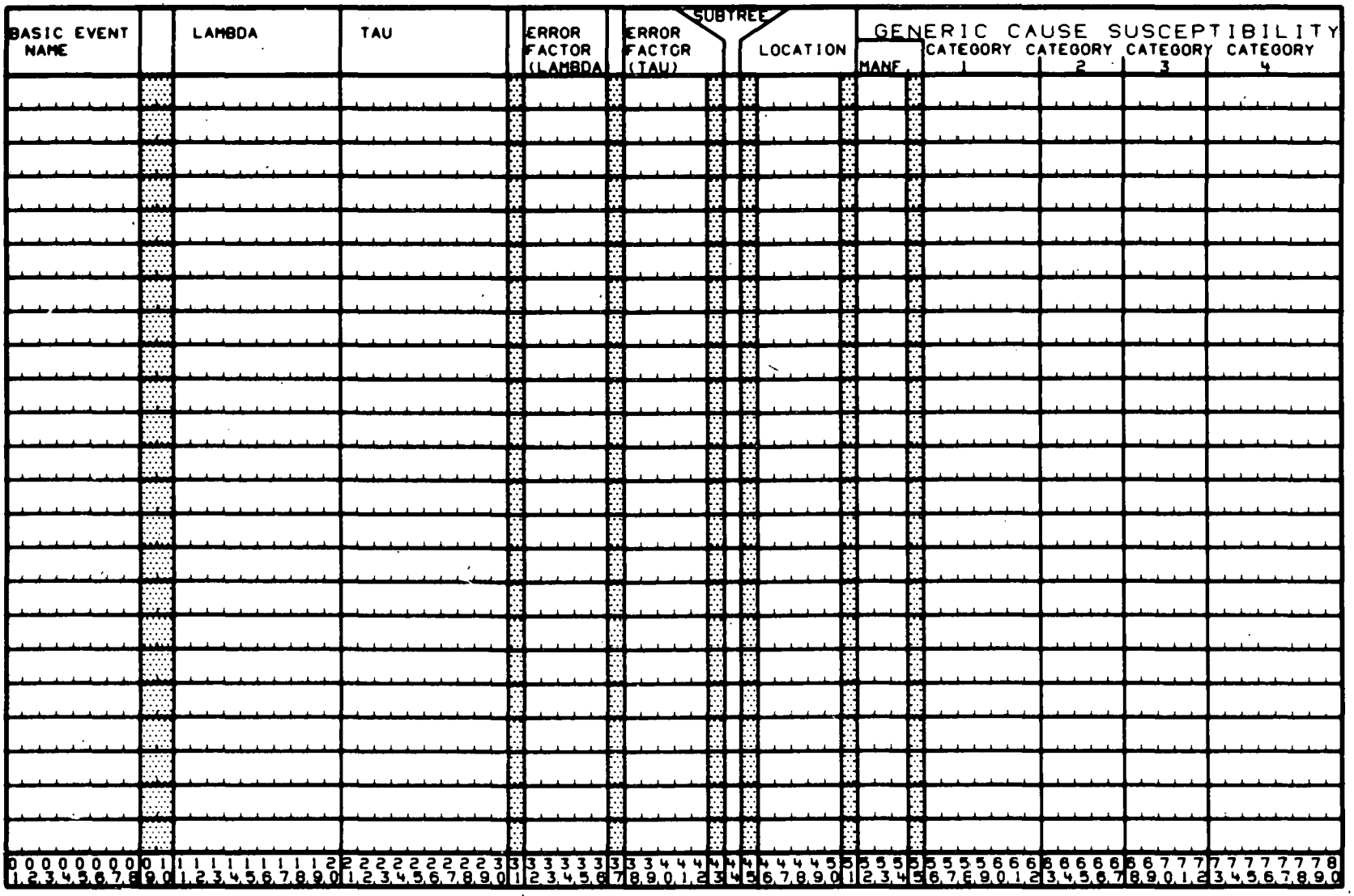


Fig. 1 Sample coding form for input to common cause analysis.

4. LOCATION

The physical location of the basic event components is the key to obtaining the prime common cause candidates from common cause candidates. Representing the location of the component in a computer code format can be a tedious task. If the analyst chooses not to use this location option, then all components are assumed to be in the same location. Prime common cause candidates are then minimal cut sets having a significant common cause event.

Basically the procedure for establishing the coding for the physical location involves coupling the component to:

(1) A specific map for a portion of the site (such as a building map)

(2) A subsection of this map (usually a room in the building)

(3) A final subdivision of each subsection of the map (usually a specific cabinet in a room).

Components that would otherwise be located in more than one subdivision (for example, a pipe running through several rooms) require a subdivision of their own. Section IV gives further details on the maps.

5. MANUFACTURER

A common manufacturer among all the basic event components in a minimal cut set renders the cut set a prime common cause candidate. This special condition is important enough to receive separate treatment. Three digits are allowed. If desired, a flag can be set to ignore this condition.

6. REPRESENTING GENERIC CAUSE SUSCEPTIBILITIES

Each basic event in the fault tree is evaluated by the analyst to determine whether it is susceptible to any generic causes in the four categories. The main generic causes are selected for each category, and the appropriate letter inserted in the coding form columns: Category 1 (Columns 56 through 62), Category 2 (Columns 63 through 67), Category 3 (Columns 68 through 72), and Category 4 (Columns 73 through 80).

The generic cause susceptibility section on the coding form (Figure 1) is split into four indicated categories with numerical labels only, and each category is divided into a fixed number of columns. The analyst can use any group of columns for any category; that is, the

category heading can be redefined as desired. That is, if chemical or miscellaneous generic causes should need more room than mechanical or thermal causes, then Categories 1 and 3 may be interchanged.

The secondary cause susceptibilities (Categories 1, 2, and 3) require only one alfanumeric character for representation. The common links given in category 4 require two alphanumeric characters for the description. The first character is used to describe the special condition (Table IV) and the second is used to indicate which set of components share the special condition. For example, the first character can denote "maintenance" and the second character indicates the set (for example, Set 2) of components that are maintained by the same individual. If all the basic event components in a minimal cut set share a special condition, no common location check is required to identify this cut set as a prime common cause candidate.

The following coding form excerpt shows both types of generic cause susceptibilities.

Generic Cause Susceptibility

	Category 1	Category 2	Category 3	Category 4
53 54 55 56	57 58 59 60 61 62	63 64 65 66 67	68 69 70 71 72	73 74 75 76 77 78 79 80
	P	R		N1

The coding form shows that the component in a particular failure mode is susceptible to pressure (Table I) and radiation (Table II), and the component is in Energy Flow Path 1 (Table IV).

IV. DETAILS ON CONSTRUCTING DOMAINS

A domain is a geographic area, divided and subdivided to reflect barriers against a particular secondary cause. Most buildings contain barriers. Walls, floors, and cabinets are common ones. An oil spill would generally be confined to the room in which the spill occurred. Vibration from a large compressor, on the other hand, may affect every room in the building. Acid vapors may become distributed through several rooms by the air conditioning system, and a maintenance error may affect the entire plant. Thus, most secondary causes have a distinct "domain" because boundaries which are capable of containing one often cannot contain another. As an example, Figure 2 represents the basic floor plan of the second floor of Building C. The rooms are labeled with their actual room numbers, and the storage cabinets in Room 206 are represented by "A" and "B". When equipment is located there, hallways also may be labeled with unique numbers.

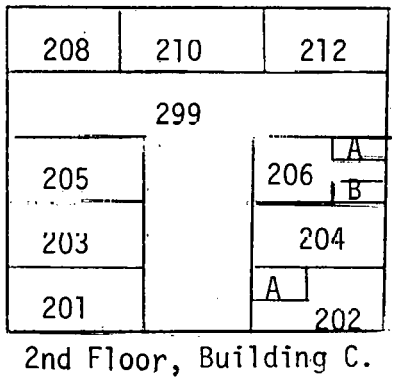


Fig. 2 Basic floor plan of second floor of Building C.

This map (Figure 2) must present the finest resolution of areas recognized in all the secondary cause domains. However for a specific secondary cause, not all the boundaries indicated by the map will necessarily be applicable. For example, the wall between Rooms 208 and 210 may be a barrier against an oil spill but not against a fire. Therefore, through use of the map, a domain is constructed for each secondary cause. A domain usually does not have as fine a resolution as a map. These domains are part of the input to the computer program. The map (Figure 2) is only an aid to the analyst during formation of the domains.

As an example of a domain from the map in Figure 2, the only barriers against "conducting medium" are Rooms 201 and 212 and Cabinet 206A. Therefore, the domain for this secondary cause is

Area 1	201
Area 2	202, 202A, 203, 204, 205, 206, 206B, 208, 210, 299
Area 3	206A
Area 4	212

In practice, every room in a building can easily be represented in a single domain and be compactly stored in the computer.

V. SAMPLE COMMON CAUSE EVALUATION

In this sample problem only one minimal cut set is considered. Figure 3 is a tabulation of all the generic cause susceptibilities for the particular minimal cut set containing basic events (B, C, D, F, H). This table would be formed internally by the computer upon determining that the combination of basic events (B, C, D, F, H) is a minimal cut set. The information given in Figure 3 can be decoded by referring to Tables I through IV.

		Generic Cause Susceptibility							
		Location		Category 1		Category 2		Category 3	Category 4
Failure Event	B	A102A		ITSG		R		A	M2
Failure Event	C	A103		IG					E1M2
Failure Event	D	A103		G		VR		A	M2C3
Failure Event	F	A100C		IG		M			M2
Failure Event	H	A103		STIG		ER			M2E1

Fig. 3 Sample cut set evaluation.

The computer selects the first generic cause susceptibility of Failure Event B: I (impact) in the mechanical or thermal category (Category 1). The other minimal cut set members are checked to determine whether this susceptibility is shared. If any minimal cut set member does not share the susceptibility, the generic cause (impact) then cannot be a significant common cause event. Basic Event D is found not to be susceptible to failure from impact. The process is repeated for all the generic cause susceptibilities of Basic Event B in the first category, comparing only within that category. Generic Cause G (grit) is found to be a significant common cause event; hence the minimal cut set is a common cause candidate. Categories 2 and 3 are searched in a similar manner; however, the search discloses no further significant common cause events. A check must now be made to determine whether the cut set is a prime common cause candidate by determining whether basic event components are in the same location with respect to grit. The domain for grit is as follows:

Area 1	A100A, A100B, A100C, A100, A102, A102A, A101, A103, A107
Area 2	A102B

Since all the basic event components are in Area 1, the cut set is a prime common cause candidate with the significant common cause event being grit.

Upon searching Category 4, "maintenance man number 2" is found to service all the components in this cut set. Therefore, the cut set is also a prime common cause candidate based on this condition.

A summary of these findings is as follows:

Prime Common Cause Candidate	Generic Cause
(B, C, D, F, H)	Grit (Significant Common Cause Event)
(B, C, D, F, H)	Maintenance (Common Link Condition)

Once the output is made available, the analyst applies that output to upgrade system safety. The analyst, aware of these common cause threats, and aided by his knowledge of the system, investigates ways to upgrade the system. He may protect the system from the grit by erecting dustproof partitions between components of the cut set, or installing grit-proof components (for example, replacing unshielded relays with those having molded casings).

To avoid the maintenance common link, special procedures may be drawn up to ensure that no single maintenance person services all components of this cut set.

The main object of this analysis is not to predict probability of failure due to common cause, but to indicate weak points in the system and suggest corrective action. For example, if a room contained a cut set of events susceptible to water, the analyst would notify cognizant personnel that every attempt should be made to eliminate water sources from that room.

VI. CONCLUSIONS AND RECOMMENDATIONS

Common cause failures are of major importance during system reliability and safety analysis. Although this report contains many new methods for the area of common cause analysis, it is, nevertheless, a preliminary report and the final computer program users manual may reflect modifications and additions to the methodology. A useful extension to these methods is adding quantification of generic cause susceptibilities. In this way the sensitivity in selecting common cause candidates can be varied. For instance, the analyst may desire only a listing of those prime common cause candidates which are strongly influenced by the appropriate common cause.

The methodology presented in this report outlines an approach amenable to a computer-aided common cause analysis that is immediately implementable and is compatible with presently used methods of reliability and safety analyses. A computer program should be developed that uses the techniques presented in this report.

VII. REFERENCES

1. E. P. Epler, "Common Mode Failure Considerations in the Design of Systems for Protection and Control", *Nuclear Safety, 10,* 1 (February 1969) pp 38-45.

2. *Reactor Safety Study: An Assessment of Accident Risks in U. S. Commercial Nuclear Power Plants,* USAEC, WASH-1400 (Draft), Appendix IV, p14.

3. J. B. Fussell, "Fault Tree Analysis - Concepts and Techniques", NATO Advanced Study Institute on Generic Techniques of System Reliability Assessment, Liverpool, England (July 1973), Nordhoff Publishing Company (1974).

4. W. E. Vesely and R. E. Narum, *PREP and KITT: Computer Codes for the Automatic Evaluation of a Fault Tree,* IN-1349 (August 1970).

5. J. B. Fussell, E. B. Henry, N. N. Marshall, *MOCUS: A Computer Program to Obtain Minimal Sets From Fault Trees,* ANCR-1156 (August 1974).

APPENDIX A

BASIC EVENT CODING INFORMATION

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

APPENDIX A

BASIC EVENT CODING INFORMATION

The following tables are used to represent the eight-character basic event identification. Table A-I gives sample system codes for an LMFBR; Table A-II the component type codes (mechanical and electrical); and Table A-III the fault mode code.

TABLE A-I

SYSTEM CODE

Code	System
A	Reactor
B	Primary Heat Transport
C	Intermediate Heat Transport
D	Steam Generator
E	Residual Heat Removal
F	Auxiliary Liquid Metal
G	Containment
I	Reactor Shutdown
J	Fuel Storage and Handling
K	Electrical Power
L	Radioactive Waste
M	Auxiliary
N	Water

TABLE A-II

COMPONENT CODE

Mechanical Components	
Code	Components
AC	Accumulator
AV	Valve, air operated
BL	Blower
CD	Control rod drive unit
CP	Pipe cap
CV	Check valve
DL	Diesel
FE	Flow element
FL	Filter or strainer
GB	Gas bottle
GK	Seal or Gasket
HE	Heat exchanger
HV	Valve, hydraulic operated
KV	Valve, solenoid operated
MV	Valve, motor operated
NZ	Nozzle
OR	Orifice
PM	Pump
PP	Pipe
PV	Pressure vessel
SV	Safety valve
TB	Turbine
TG	Tubing
TK	Tank
TZ	Transmitter
VT	Vent
VV	Valve, relief
XV	Valve, manual

TABLE A-II (contd.)

Electrical Components	
Code	Components
AM	Amplifier
AN	Annunciator
AS	Buffer
BC	Battery charger
BS	Bus
BY	Battery
RE	Relay
CL	Actuator controller
CA	Cable
CB	Circuit breaker
CC	Capacitor
CM	Comparator
CN	Converter
CT	Current transformer
DC	dc Power supply
DE	Diode or rectifier
FU	Fuse
GE	Generator
GS	Ground switch
HG	Heating element
HT	Heat tracing
IV	Inverter
KS	Lockout switch
LA	Lightning arrester
LS	Limit switch
LT	Light
ME	Meter
MO	Motor
MS	Motor starter
ND	Neutron detector

TABLE A-II (contd.)

Electrical Components	
Code	Components
OT	Transformer
PS	Pressure switch
PT	Potentiometer
QS	Torque switch
lN	Transistor
RS	Resistor
IC	Integrated circuit
SW	Hand switch
TC	Temperature controller
TF	Transmitter
TI	Timer
TM	Terminal board
WR	Wire

TABLE A-III

FAULT MODE CODE

Code	Fault Mode
A	Does not start
B	Open circuit
C	Closed valve
D	Does not open
E	Engaged
F	Loss of function
G	Disengaged
H	Optional for the analyst
I	
J	Short across
K	Does not close
L	Leakage
M	Exceeds limit
N	No input
O	Open valve
P	Plugged
Q	Short to power
R	Rupture
S	Short to ground
T	Optional for the analyst
U	
V	
W	Does not actuate
X	Operational or Maintenance fault
Y	Optional for the analyst
Z	

REPORT III

A LIBRARY FOR PRESERVING COMPONENT FAILURE LOGIC INFORMATION

J. C. Zipperer
J. B. Fussell
G. R. Burdick
J. R. Wilson

ABSTRACT

A method for storing component failure logic information for use as an aid to system failure logic modeling during reliability and safety analyses is presented. This information, in the form of mini fault trees, is system independent and can be used repeatedly. The use of this cataloged information is a first step in standardizing system failure logic modeling, such as fault tree construction.

CONTENTS FOR REPORT III

FIGURES

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

A LIBRARY FOR PRESERVING COMPONENT FAILURE LOGIC INFORMATION

I. INTRODUCTION

Mini fault trees (MFT) are the basic segments that are used to construct system logic models during reliability and safety analyses. Each MFT describes one mode of failure for a component and is modeled using Boolean failure logic along with other information. The concept of MFT offers the first step in an effort to standardize system logic modeling, because it lends itself to cataloging in a central library.

The purpose of a MFT library is to provide analysts a vehicle for pooling system independent knowledge about specific component malfunctions. Analysis of components, which is carried out during all system reliability and safety analyses, is then not lost but rather is made available for future reference by all analysts. The scheme presented here, when implemented, is compatible with other efforts to catalog information concerning components such as the Nuclear Plant Reliability Data (NPRD) bank[1].

The concept of the MFT is not new. The MFT was introduced at the Georgia Institute of Technology in 1972 by Fussell[2] as component failure transfer functions. Independently and shortly afterward, Professor G. J. Powers, at that time at the Massachusetts Institute of Technology, recognized the need for these building blocks for system failure logic models. Powers called them mini fault trees[3]. Both Powers and Fussell developed their concepts in conjunction with computer-aided logic model construction. Nielsen[4] and Taylor[5] also developed a similar basic structure for use in cause-consequence analysis. At meetings in Liverpool in July 1973, and in Berkeley in September 1974, MFT as an aid to practical construction of logic models were discussed by R. A. Evans, J. B. Fussell, D. S. Nielsen, G. J. Powers, J. R. Taylor, and W. E. Vesely.

Section II of this report is concerned with basic definitions and concepts. Section III provides an elementary example that illustrates a great many of the concepts of the MFT. In Section IV the use of MFT during logic model construction is addressed. In Section V a more detailed example is developed that illustrates the procedural steps involved in the construction of MFT. Sections VI and VII give examples of MFT as encountered in practice. Conclusions and recommendations are given in Section VII.

II. DEFINITIVE CONCEPTS

The MFT library is a collection of the sets of MFT for individual components. In general, the number of MFT per set is the number of failure modes of the component multiplied by the number of possible nonfailed states of the component. A nonfailed state of a component is a configuration the component can take on during its functional life. For example, a valve can be open or closed and a motor can be on or off.

Information contained in the MFT library includes:

(1) Boolean failure logic concerning each mode of failure of the component

(2) Secondary cause susceptibility

(3) Parts of the component that fail with relatively high frequency.

The Boolean failure logic of the MFT may consist of as many as seven parts. All of these parts can be determined from the fundamental workings of the component isolated from any system environment:

(1) The "output event" is the mode of failure being considered. For a particular component, more than one MFT may exist with the same output event, depending on the "coordinator" to be defined later.

(2) The "output logic gate" designates the logic with which the MFT is coupled into the logic model with other appropriate MFT having the same output event. One output logic gate is associated with each MFT.

(3) "Internal events" are fault events requiring further logical development within the MFT.

(4) "Internal logic gates" designate the logical development of the internal events as required by the output and input events.

(5) "Input events" can be either basic events or undeveloped fault events. Input events represent the furthest development of the output event possible by considering the isolated component.

(6) The "discriminator" is a flag designating which MFT may coexist in the final logic model. The discriminator can be determined from the component because it indicates which output events can actually coexist within the same component. The discriminator does not appear anywhere in the final system logic model.

(7) The "coordinator" is a flag indicating which MFT in a given set is to be used in the logic model. The coordinator depends on the component initial condition, such as contacts open or valve closed. For a given output event, several MFT may exist. The one applicable during construction of a particular logic model depends on the initial condition of the component.

The concept of the Boolean failure logic of the MFT is given in Figure 1.

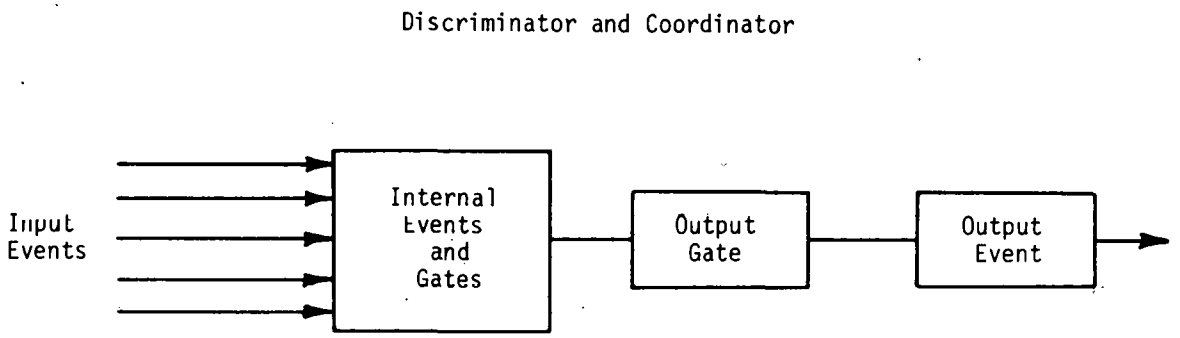


Fig. 1 Concept of mini fault tree.

The tabulation of secondary cause susceptibility is a listing of environmental factors that accelerate mechanisms of failure or cause instantaneous failure of the component. Examples of secondary causes that might be included in this tabulation include dust, shock, corrosion, and maintenance errors.

A listing of the parts of the component that fail with relatively high frequency is useful when the resolution of the analysis is extended beyond the component in question. The component failure then becomes an event in the logic model and the development is carried out to basic events reflecting malfunction causes of the parts.

A portion of a MFT for a component is determined by conventional failure mode analysis. Failure mode analysis is a method of identifying all possible means by which a device can fail to perform its required function. This failure mode analysis then immediately provides the MFT output event. The output logic gate is determined by recognizing the logical relation the device failure has to the output event. That is, the output logic gate depicts the way the event being developed is transferred through the component. If the component failure alone can cause the output event being developed then the output logic gate is OR. If, however, the component failure is required in addition to the fault event being developed, then the output logic gate is AND. Internal events give further information about the failure mode. Their appearance in a final fault tree gives a logical relationship between the internal events and the input events. The input events are primary failures or fault events. After this failure mode analysis information has been incorporated into the set of MFT, the discriminator is set and is generally determinable from the output event description. The coordinator influences the event descriptions appearing in the MFT. For example, if contacts are initially open then the failure modes resulting in closed contacts must be stated as the "contacts transfer closed". On the other hand, if the contacts are initially closed the modes are stated as "contacts fail closed".

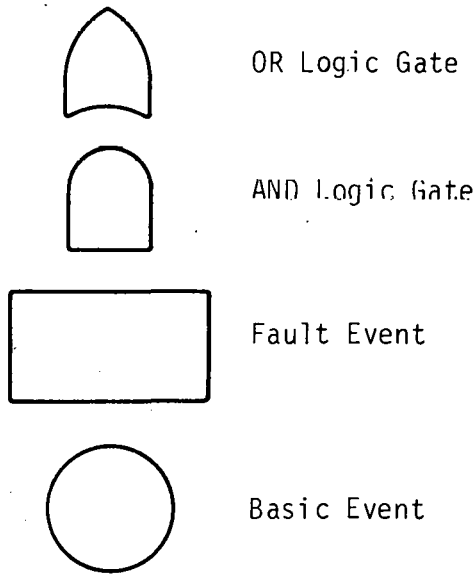
III. AN ELEMENTARY EXAMPLE

As an elementary example of MFT development, a fuse is chosen as the component of interest. This example is meant only to illustrate the concepts already presented.

To determine MFT for a fuse, the ways a fuse can fail are determined by considering the design of the fuse: A fuse is an over-current protective device, with a circuit-opening fusible member directly heated and destroyed by overcurrent. A fuse by not performing as intended can fail by transmitting an overload. Also, since the fusible member of a fuse transmits current under normal operation, the fuse can also fail so as to cause "no current". These then denote two MFT for a fuse, one with the output event "overload" and another with the output event "no current". Since the fuse has only one configuration, as opposed to a switch that can be open or closed and not be failed, the coordinator is not necessary. The MFT for the output event "overload" will be determined first. Since the fuse alone cannot cause an overload – it can only transmit an existing overload -- the output gate is AND. Only one event is input to the MFT: basic failure of the fuse to open circuit when subjected to an overload. No internal events or logic gates occur. For the output event "no current", the output gate is OR because the fuse alone can cause "no current". Again only one input event occurs – basic failure of the fuse (fuse opens).

The MFT for a fuse are given in Figures 2 and 3. The discriminators are different so as to denote that the two MFT output events are not allowed to coexist; that is, a fuse cannot be failed open and closed at the same time.

The Boolean logic symbols used herein are:



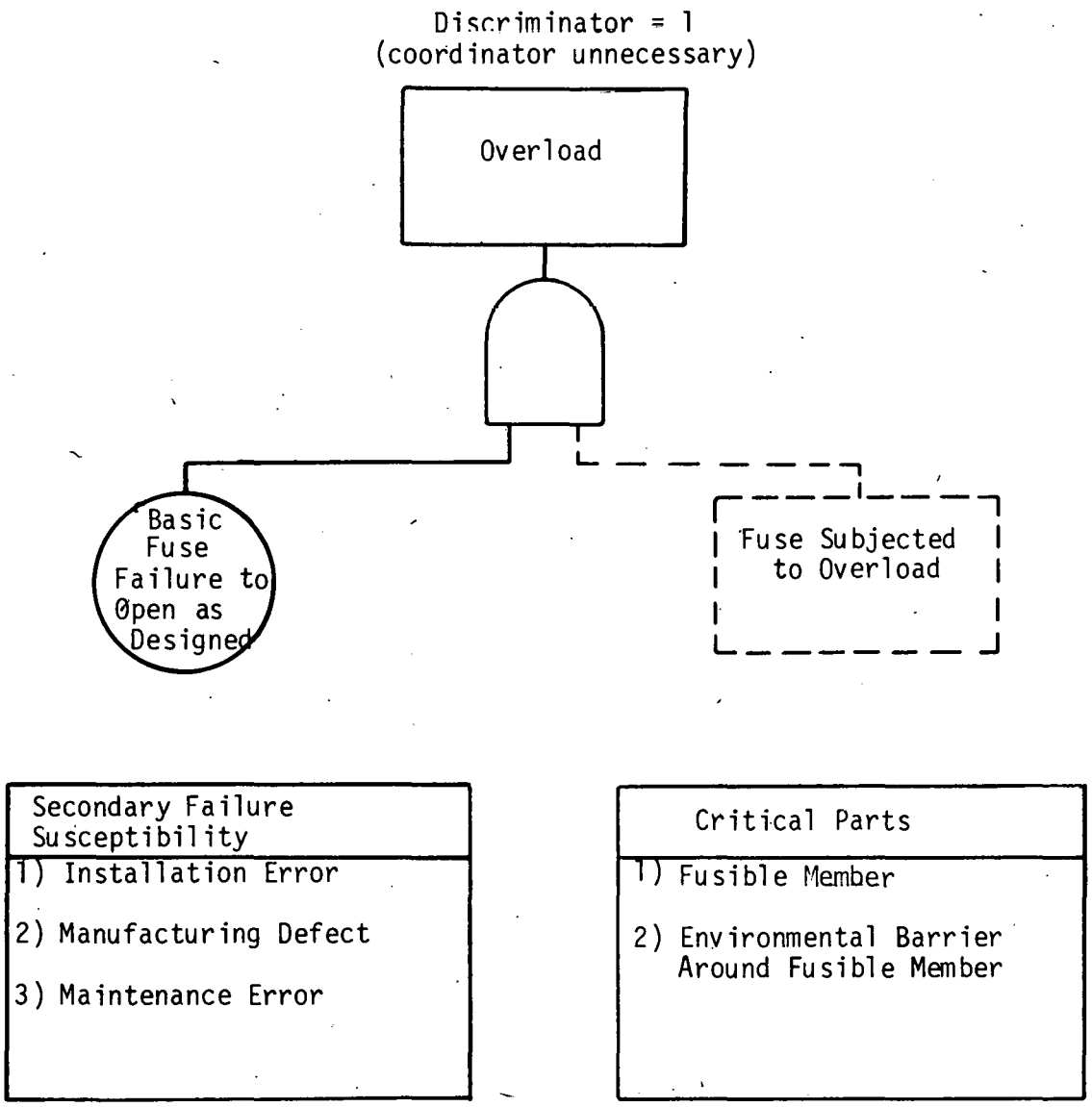


Fig. 2 MFT for a fuse with overload as the output event.

Discriminator = 2
(Coordinator Unnecessary)

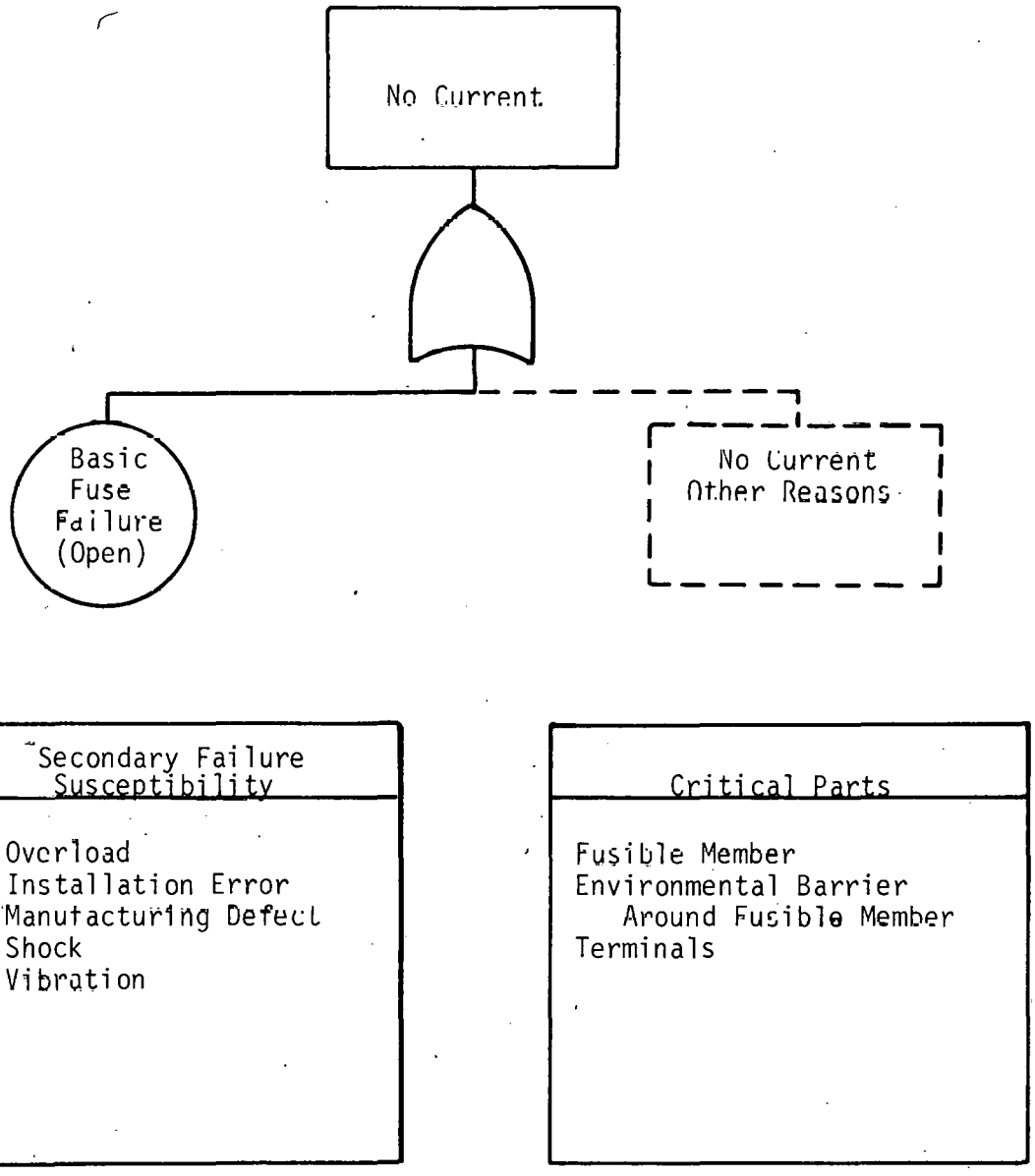


Fig. 3 MFT for a fuse with no current as the output event.

The output of an OR logic gate exists if one or more of the inputs exist. The output of an AND logic gate exists only if all input coexists.

During applications of the MFT, the resolution of the analysis can be extended in two ways. The basic event (circle symbol) can be replaced by a corresponding fault event (rectangle symbol) and the logical development can be extended to the part level. In which case, the basic events reflect part malfunctions. The resolution is also extended by logically developing possible secondary causes of failure in addition to the basic failure. The specific secondary causes to be developed are system dependent. Tabulation of secondary cause susceptibility is an aid to stimulate thinking with regard to possible secondary causes.

In Figures 2 and 3 the events "fuse subjected to overload" and "no current (other reasons)", respectively, are flag events that indicate system conditions that must be considered during system logic model construction. This feature will be illustrated in the next section.

IV. USE OF MFT IN LOGIC MODEL CONSTRUCTION

MFT are useful when the logic model construction process has reached the level of system components causing subsystem faults. At this level the analysis is engaged in deducing causes for subsystem faults. Examples are "insufficient flow in a particular flow loop" or "overcurrent in a specified electrical circuit". These subsystem faults are called "second order fault events" (SOFE).

Also, MFT are used when the level of construction has reached the point that a component is directly causing another component to malfunction (command fault). Examples are "relay coil holds relay contacts open" and "engine fails to provide power to generator". These faults are called "fourth order fault events" (FOFE). Details on ordered fault events are given in Reference 2.

A simple example is given here to illustrate how MFT are used to develop SOFE in a system. Figure 4 is a schematic of the sample system. Reference 2 gives details on the use of MFT.

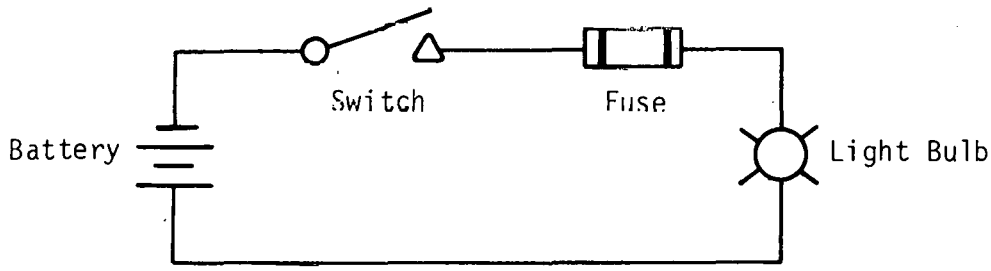


Fig. 4 Sample system.

The TOP event is "no light from the bulb". The switch is initially closed and wiring failures are neglected. The tree top to be developed is shown in Figure 5.

The components in the circuit that can possibly contribute to the fault, no current, are (a) fuse, (b) switch, and (c) battery. The light bulb has been treated in the tree top so it need not be considered again. The MFT library is then consulted for the MFT of the components of interest with the correct output event, no current, and the correct coordinator. The coordinator is pertinent only to the switch because it is the only component with two possible operational states. The appropriate MFT are then collected and are shown in Figures 3, 6, and 7.

Since all the output gates are OR logic gates, an OR logic gate is used to develop the fault event, "no current in circuit", and the order of input to the OR logic gate is immaterial. The MFT are then added to the tree top as shown in Figure 8. The dummy event "no current (other reasons)" is dropped from the MFT of the battery because it is the last appropriate MFT considered. Figure 8 is given only for tutorial purposes; the actual representation that would be used in practice is given in Figure 9.

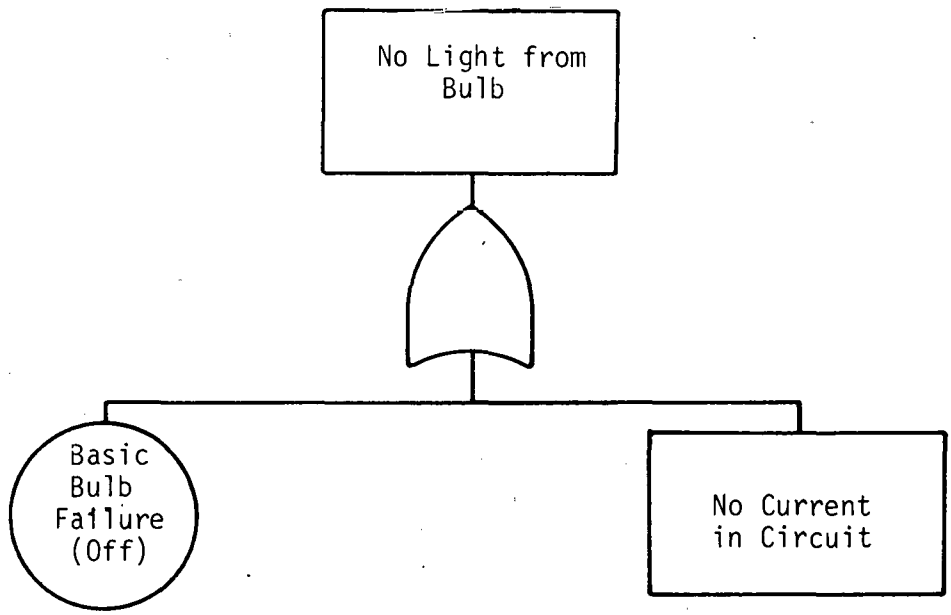


Fig. 5 Tree top of sample system.

The event "switch opened" is a fourth order fault event, but since a system boundary has been reached, this event is shown in a diamond symbol in Figure 9 and not developed further.

The logic model in Figure 9 is complete to the level of basic failures. Any one of the basic failures could be developed further with secondary causes such as those given in the tabulation in the MFT.

Discriminator = 2

Coordinator = Switch Initially Closed

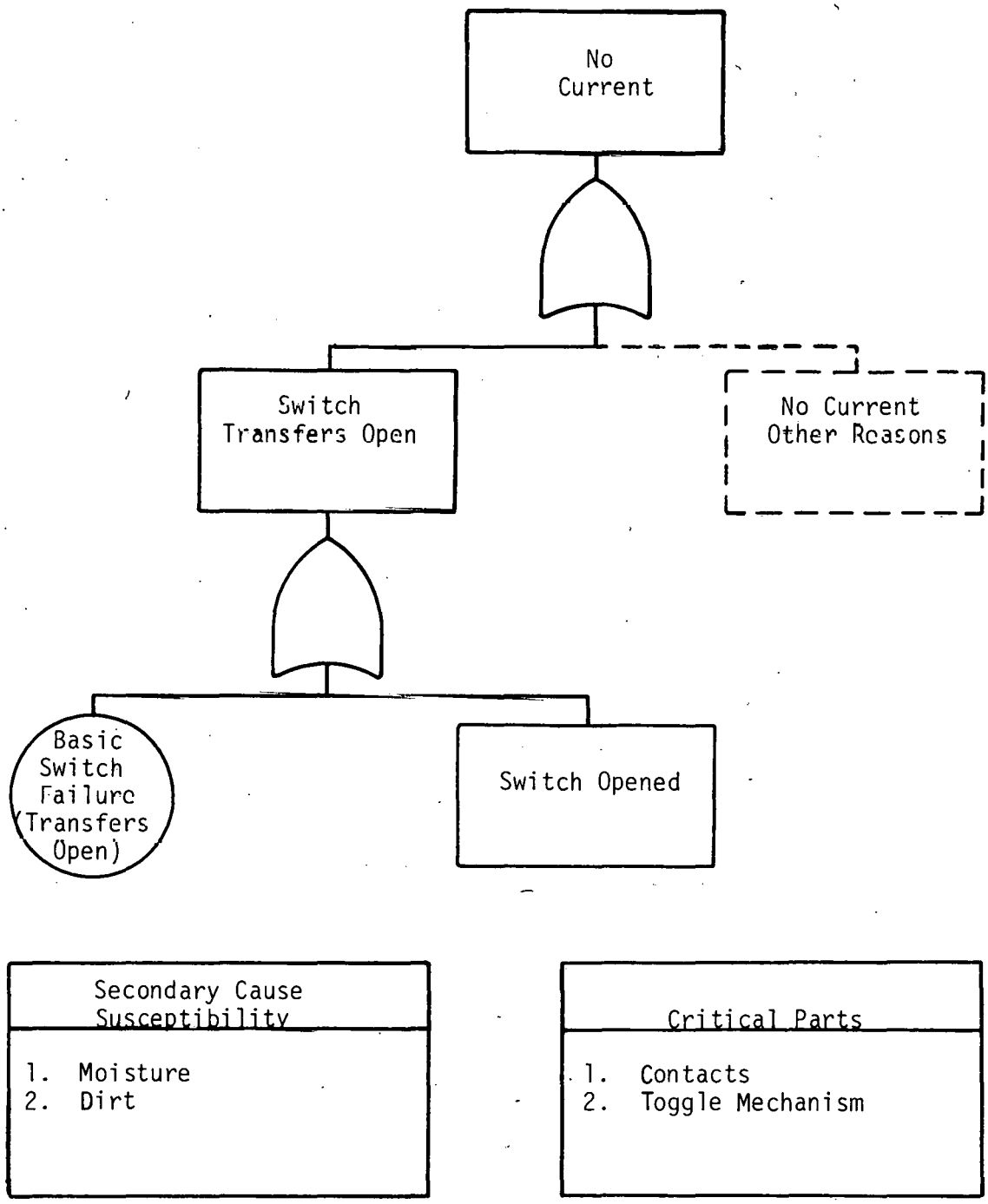


Fig. 6 Applicable MFT for a switch.

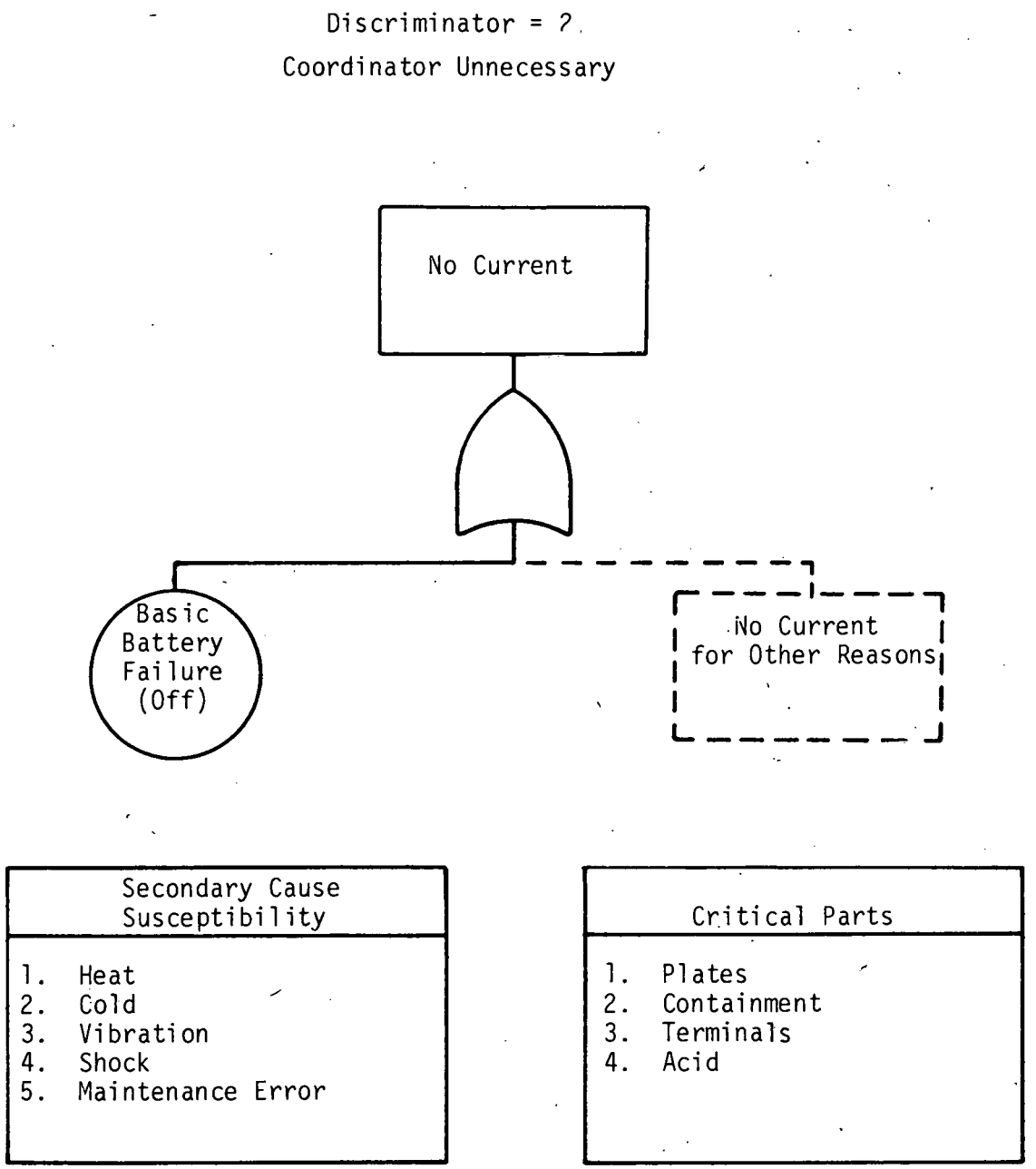


Fig. 7 Applicable MFT for a battery.

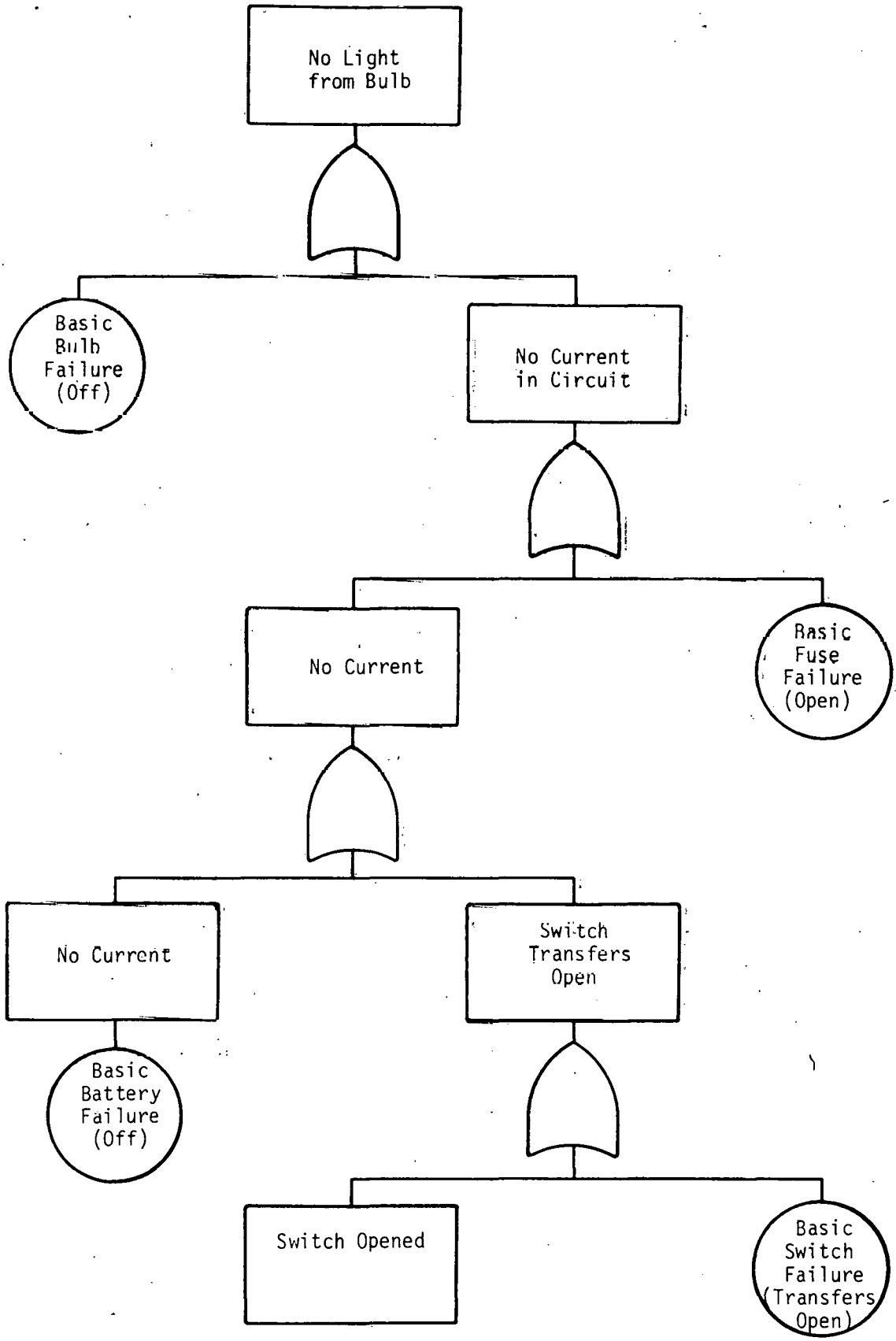


Fig. 8 Tutorial step in logic construction using MFT.

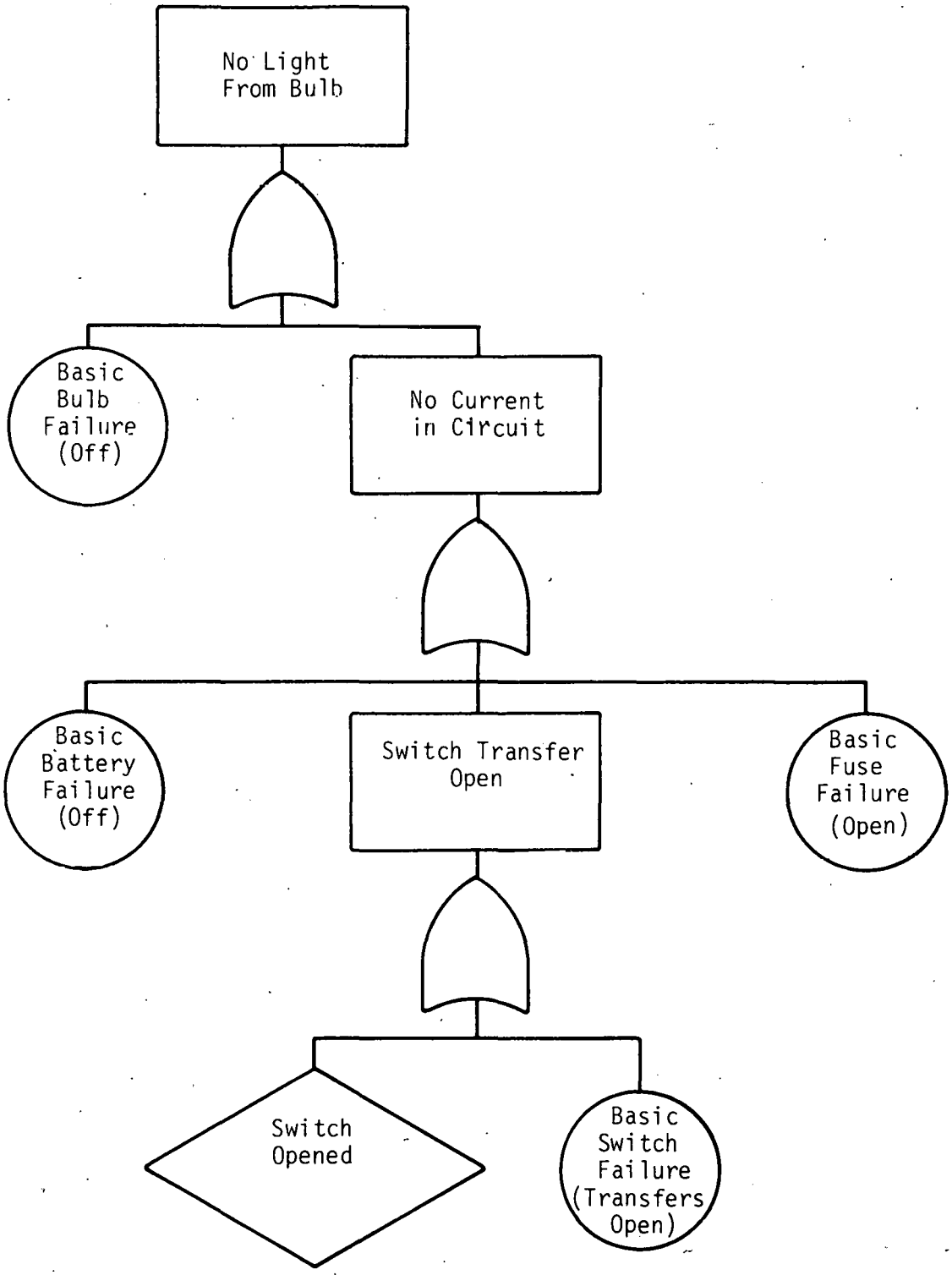


Fig. 9 Logic model representation using MFT for development.

V. PROCEDURAL EXAMPLE

This example is provided to illustrate steps the analyst could logically follow in constructing MFT. A free surface sodium pump is used as the example. However, the construction procedure is independent of the pump. All components require basically the same procedure.

1. COMPONENT RESEARCH

The analyst must become acquainted with the physical characteristics and design purpose of the component. From these he may deduce many possible sources of component malfunction. In addition, there are available from private and public sources case histories of component malfunctions which have actually occurred in operating nuclear power plants. These case histories are available, for example, from the Nuclear Safety Information Center, P. O. Box Y, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37830. Such histories are a valuable, although not absolutely necessary, aid in eliminating oversights and omissions regarding sources of component malfunction. They are particularly useful in pointing out sources of secondary faults and failures.

In the case of this example, the analyst has found that the free surface sodium pump is a design of the centrifugal type. It is utilized as the prime circulator in the primary and secondary cooling loops of sodium cooled reactors. Figures 10 through 13 are representative of the various designs. This type of pump maintains a "safe" sodium level in the pump housing through an intricate balance of wear ring, weep hole, cover gas pressure, and sodium return line design characteristics. The free surface sodium pump has no internal seals or internal lubricated bearings. The bearings and seals are located on top of the pump floor plate. An unlubricated hydrodynamic guide bearing, located inside the pump, guides the impeller shaft.

The Liquid Metal Engineering Center reports in Reference 6 the following critical characteristics, predominent failure modes, conditions, and mechanisms observed in their research.

Critical Characteristics Observed

(1) The alignment and life of bearings, seals, and shafts.

(2) Adequate case cooling to prevent thermal distortion during startup or transients.

(3) Sodium level control.

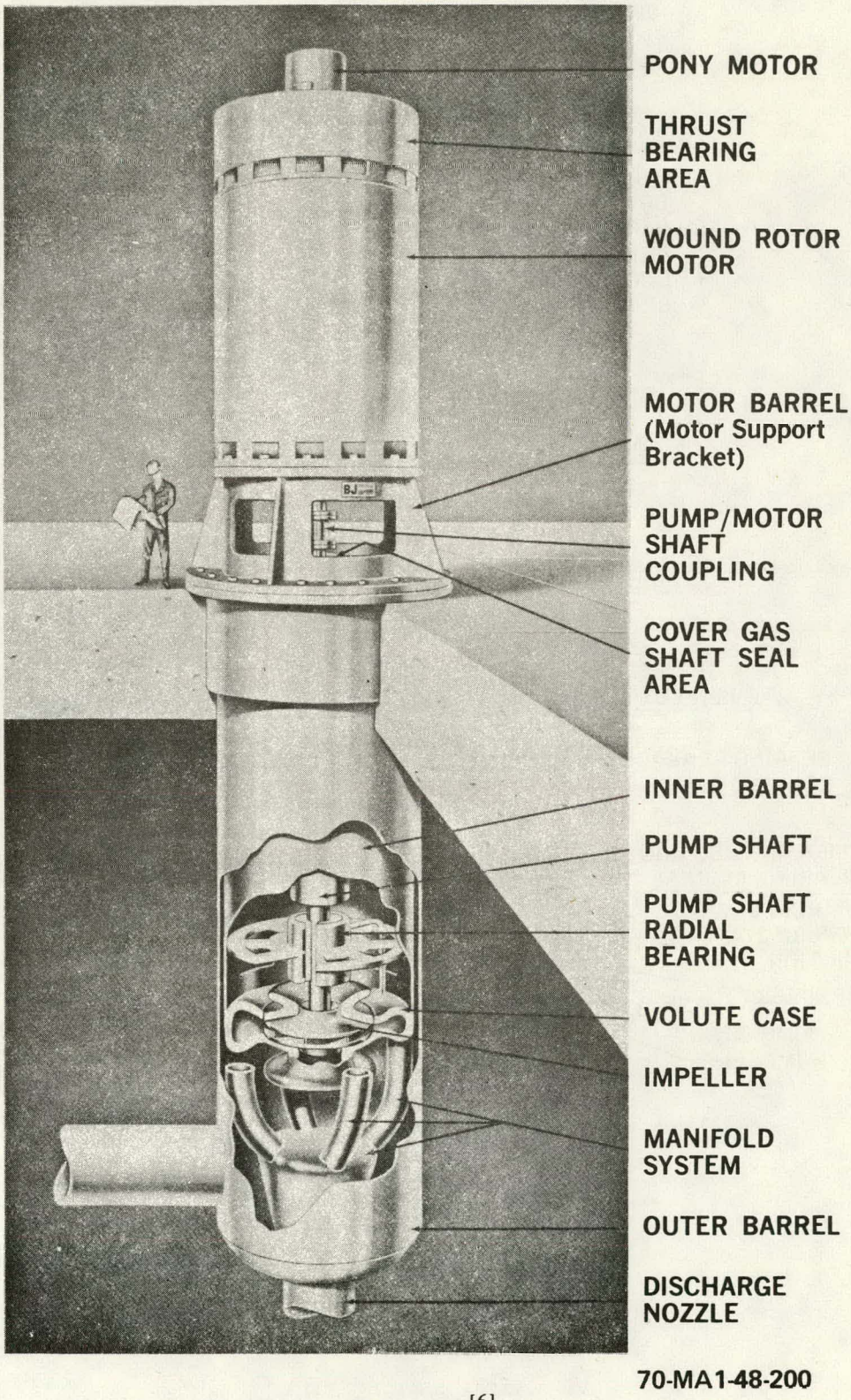


Fig. 10 Pump and drive concept[6]

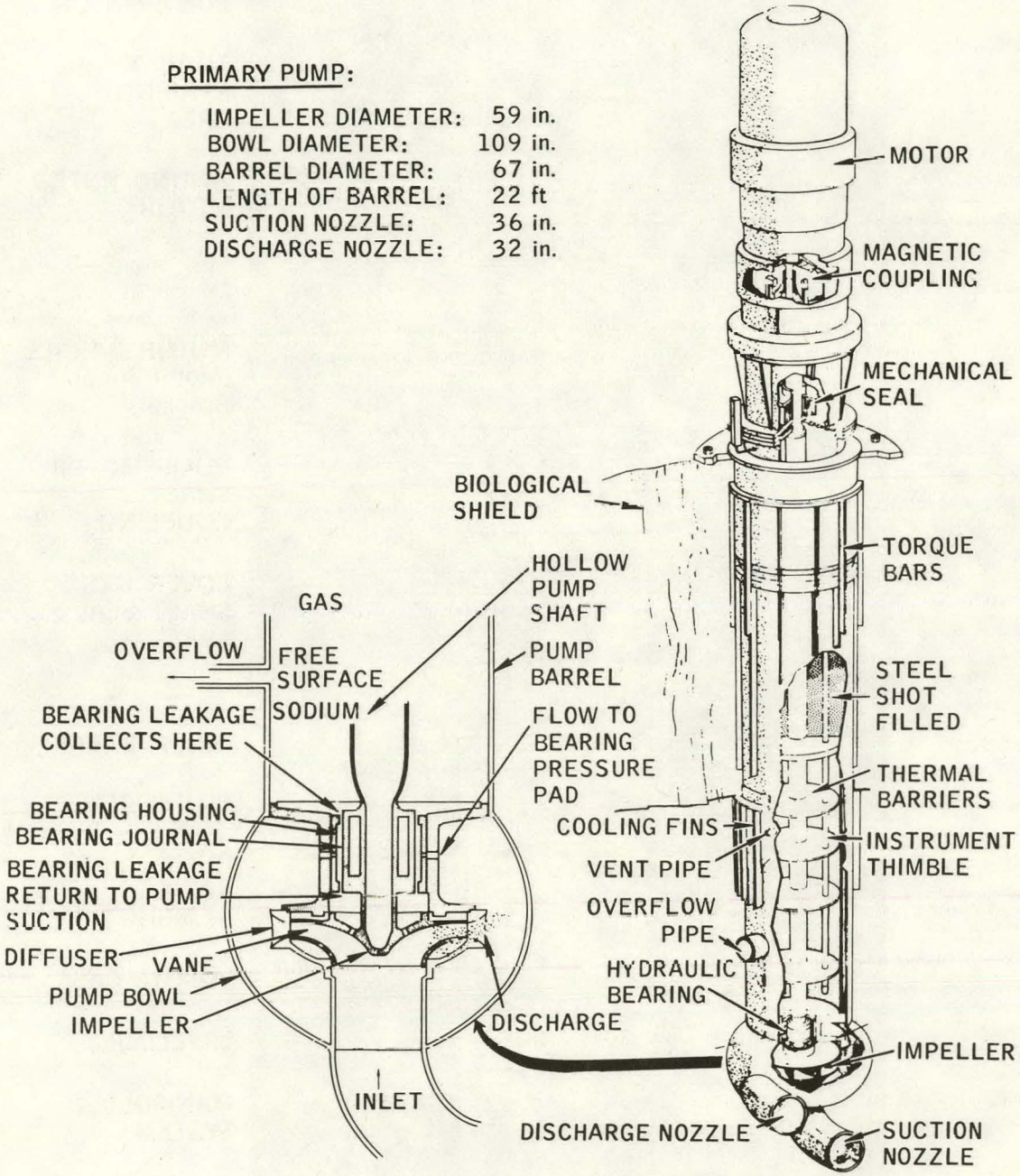


Fig. 11 Free surface sodium pump[6].

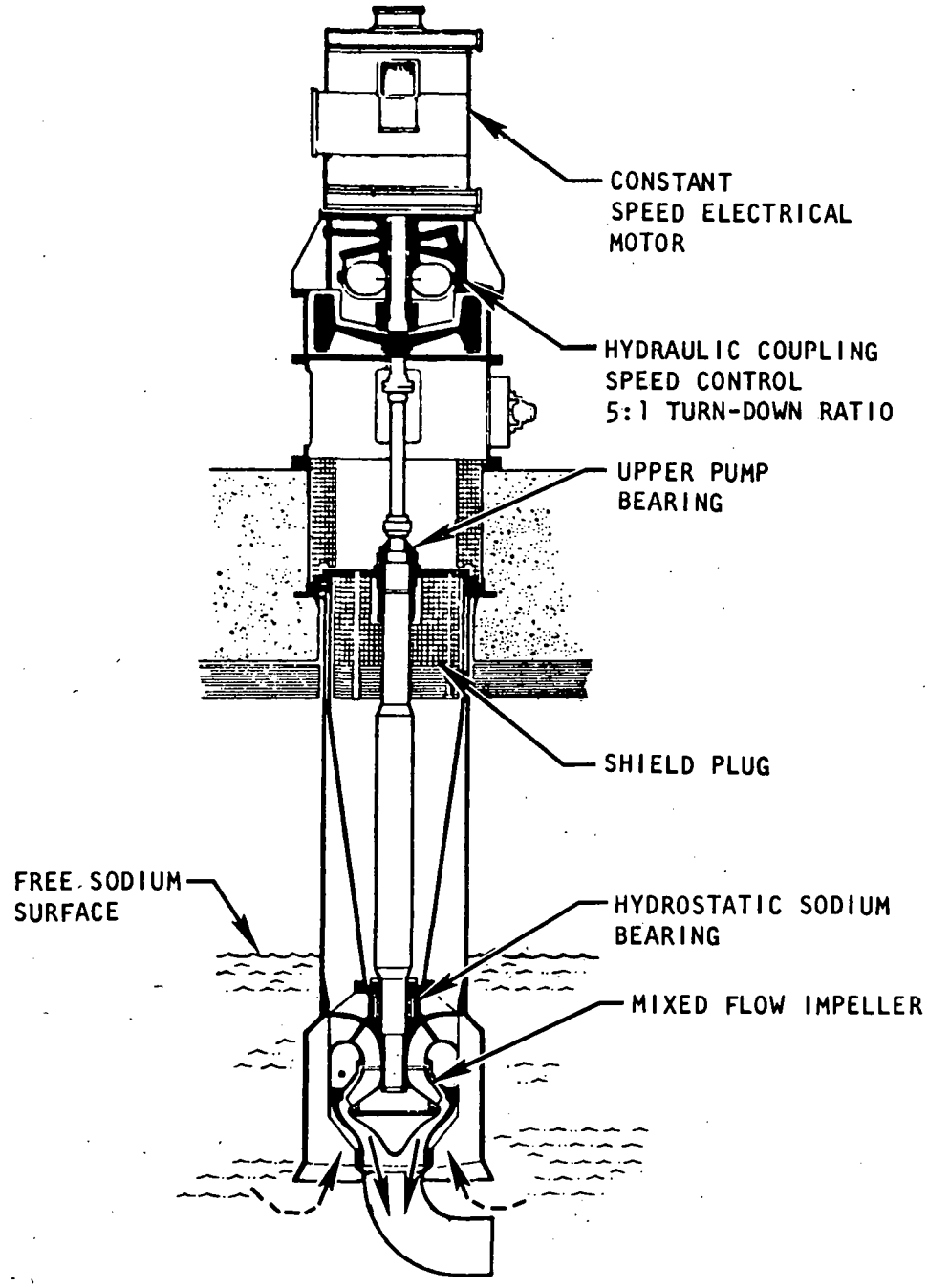


Fig. 12 PFR primary sodium pump[6].

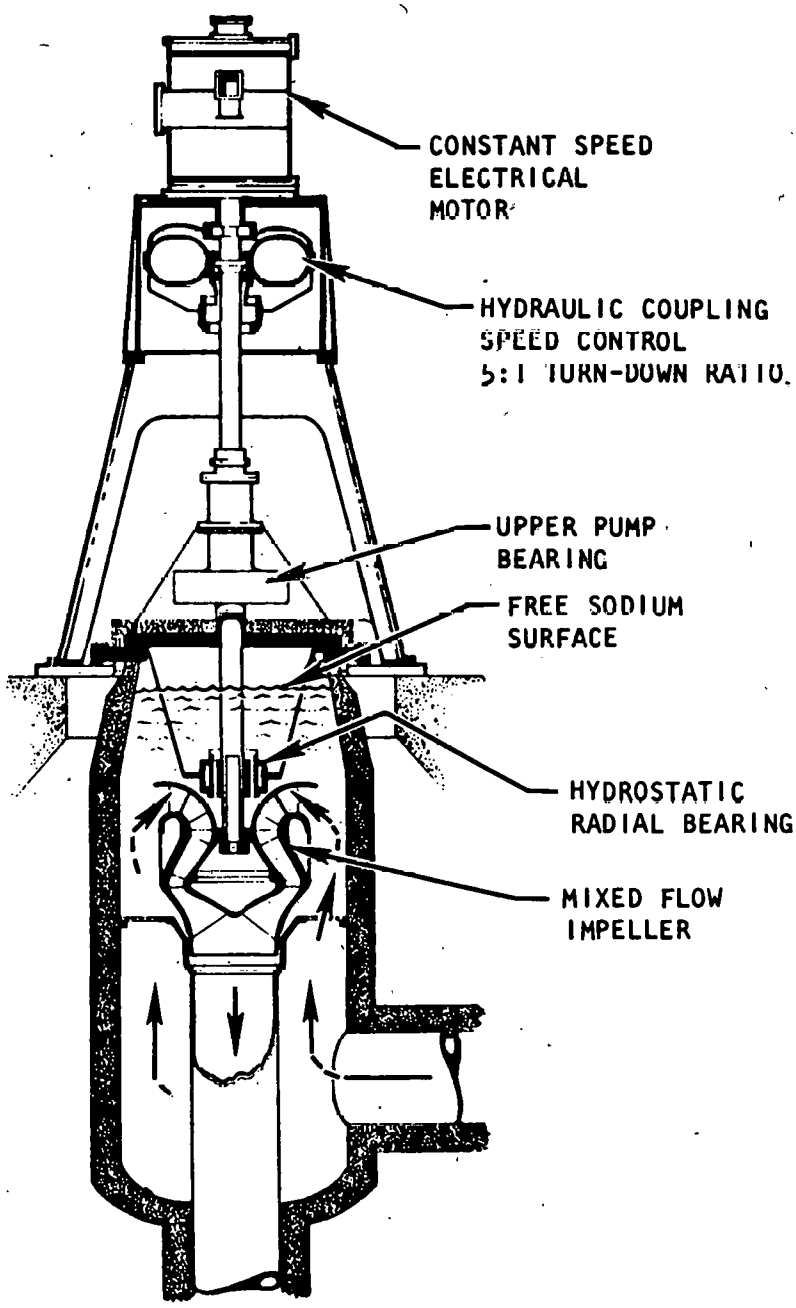


Fig. 13 PFR secondary sodium pump[6].

Failure Modes Observed

(1) Restricting flow in system.

Failure Conditions Observed

(1) Misalignment of shaft seals.

(2) Vibration of bearing.

(3) Foreign material in rotating seal.

(4) Oil leaking into sodium from oil seal.

(5) Weep holes sucking sodium into pump casing.

(6) Shaft deflection.

(7) Sodium on pump shaft leaking into oil reservoir.

Failure Mechanisms Observed

(1) Bearing wear.

(2) Shaft seal wear.

(3) Carbon face seal rings chipping.

(4) O-ring wear.

(5) Shaft seizure due to foreign material.

(6) Binding due to shaft distortion from thermal gradients.

(7) Piston cup wear.

Problems Identified

The following are some of the more serious problems encountered with free-surface pumps:

(1) Preheating the system prior to filling with sodium has caused warpage that resulted in binding the impeller shaft.

(2) Foreign matter (metal cuttings) have deposited in the hydrodynamic bearing and caused seizure of the impeller shaft.

(3) An impeller shaft was bent during the installation of the bearing seal assembly on the pump floor plate resulting in loss of plant availability.

(4) Mechanical oil seals located in the bearing seal assembly (top of pump floor plate) have failed resulting in downtime.

(5) Mechanical oil seals have failed resulting in oil leakage into the sodium system. This is an unacceptable incident and the bearing seal assembly must be designed to make oil leakage into the sodium system impossible.

(6) Magnetic clutches used for driving some free-surface pumps have operated at higher than specified temperatures as the result of inadequate air circulation inside buildings.

(7) Problems with belt-driven tachometers have resulted in unnecessary shutdowns.

(8) Permitting pump drive motors to be exposed to the elements (rain) has caused motor failure resulting in downtime.

(9) The lack of adequate operator training has resulted in downtime, sometimes due to oil getting into the sodium system.

(10) The impeller shaft guide bearing journal has been scored in some pumps; however, pump failure has not ensued.

(11) Bearings have failed in the bearing seal assembly on some pumps; however, other pumps have operated for thousands of hours without bearing problems.

(12) Pumps equipped with balancing legs have been flooded by improper operation of inert cover gas systems, resulting in downtime and plugged vent systems.

(13) Carbon (oil in sodium system) has deposited in bellows, actuated level indicators on the pump cases, and rendered them inoperative, resulting in pump downtime.

(14) Pump downtime has resulted because of pump instrument thimbles vibrating. The thimbles were attached on the upper end only.

Using the preceding information, the analyst may deduce causes of component malfunction and incorporate these in the MFT.

2. ENGINEERING DATA PREPARATION

If additional information is desired, the analyst may have to list specifications and parameters that characterize the component because computer printouts of failure histories

are often keyed to such specifics and ranges of the parameters. Following the NPRD format[1], the analyst has prepared, for the free surface sodium pump of this example, the following table of engineering data:

Westinghouse Electric Corporation Primary Coolant Pump at FFTF

A	Component/NPRD Code Free Surface Sodium Pump/PUMPFN
B	Type/NPRD Code Centrifugal/B
C	Inlet Size/NPRD Code 28 in./E
D	Materials (Body)/NPRD Code 304 Stainless/B
E	Type of Shaft Seal/NPRD Code Mechanical Seal/B
F	Flow Capacity/NPRD Code 10,000 - 50,000 gpm/D
G	Total Developed Head at Rated Capacity 500 ft
H	Flow Rating 14,500 gpm

The research of Section IV-1 has provided information concerning design requirements and functional objectives of the component. Excursions from these limits define the output events for the MFT. The analyst organizes this information as follows:

Free Surface Sodium Pump

Component Functional Objective

To transfer 14,500 gpm at 500 ft sodium head and 1,050°F.

Functional Parameters Necessary

H = head - ft = f (P, V, Z, w, g)

P = pressure - psi

V = flow velocity - ft/sec

Z = static head - ft

w = specific weight - nondimensional

g = gravitational acceleration - ft/sec^2

Q = flow rate - gal/min

t = temperature - ^oF

Failure Criteria

$Q > Q_U$

$Q < Q_L$

$H > H_U$

$H < H_L$

where

U = upper system limit

L = lower system limit

3. DEVELOPING A COMPONENT FUNCTIONAL BLOCK DIAGRAM

As an additional aid in developing the MFT, the analyst identifies all basic functional input and output parameters plus any environmental parameters that may affect proper operation of the component. These parameters are as follows:

(1) Input control parameters

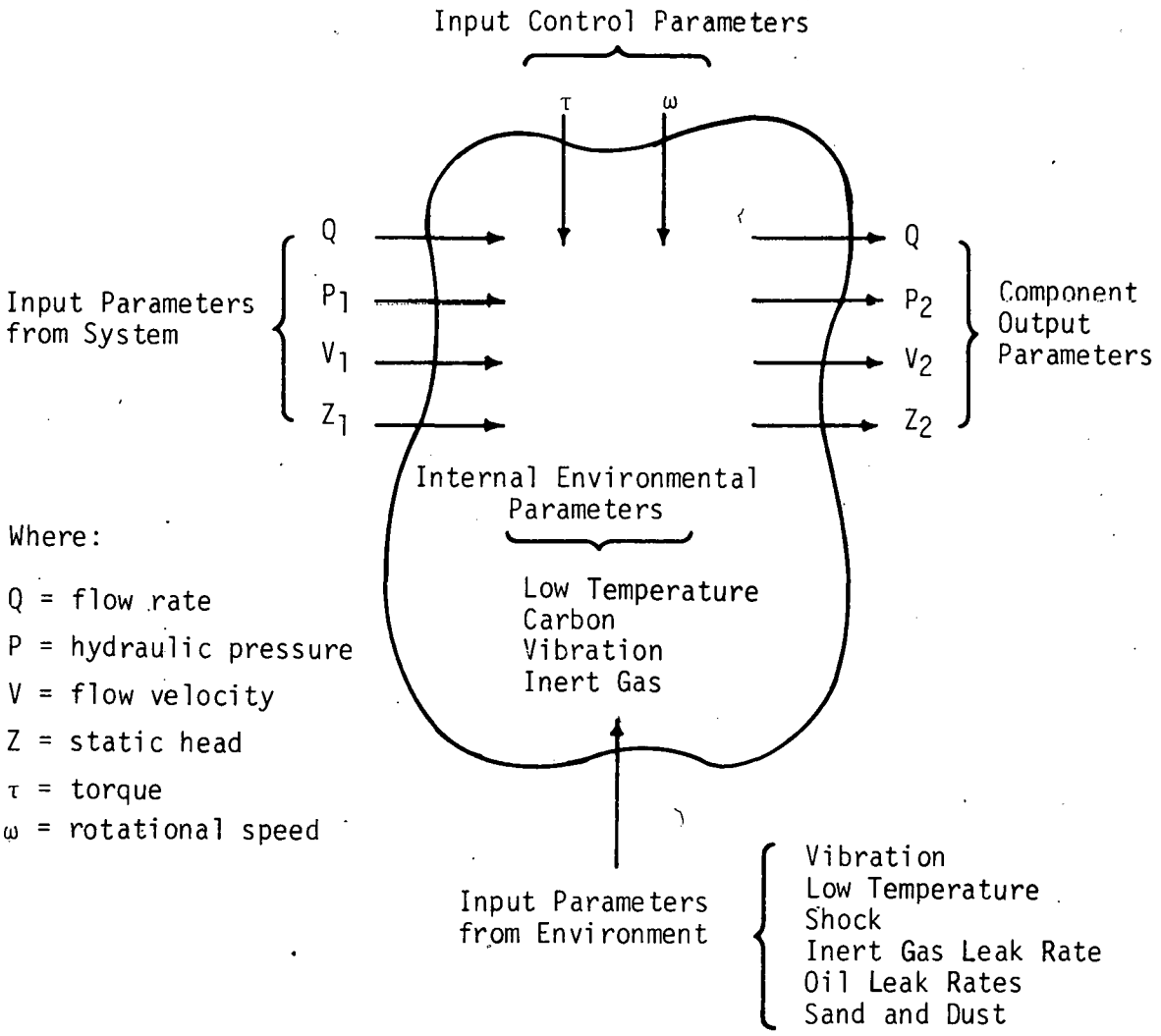
(2) Input parameters from system

(3) Input parameters from environment

(4) Component output parameters

(5) Internal environmental parameters.

From these parameters the analyst constructs a functional block diagram which illustrates in simple form the various inputs to and outputs from the component as shown in the following illustration.



Where:

Q = flow rate
P = hydraulic pressure
V = flow velocity
Z = static head
τ = torque
ω = rotational speed

4. DEVELOPING THE MINI FAULT TREES

The analyst is now prepared to construct a set of MFT for the free surface sodium pump. In doing so, the following steps might be followed:

(1) The states of the component are identified. For example, the pump can be operating or not operating.

(2) The possible output events are identified. For example, pump malfunction can result in (a) inadequate head, (b) inadequate flow, (c) excessive head, or (d) excessive flow.

(3) The component output logic gate is determined to be an AND or an OR gate. Since the pump alone can cause all the above listed output events, the output logic gate is in every case an OR gate.

(4) The internal failure logic of the MFT is determined for each output event.

(5) The input events to each MFT are identified. A basic component failure (circle) is not necessarily required. If appropriate, a command fault should be indicated.

(6) The secondary cause susceptibility is determined for each MFT.

(7) The critical parts for each MFT are determined.

(8) The discriminator is set for each MFT. Different discriminators are set for output events that cannot exist simultaneously, not for output events that are not expected to exist simultaneously.

(9) The information is coalesced into the least possible number of MFT for each component.

In the following discussion, an example of a set of MFT for a free surface sodium pump (Figure 14) is given as it might appear in the library. The output events "inadequate head" and "inadequate flow rate" are coalesced because the MFT are identical. Also, "excessive head" and "excessive flow rates" are coalesced. In the final analysis, the MFT for this pump could be updated.

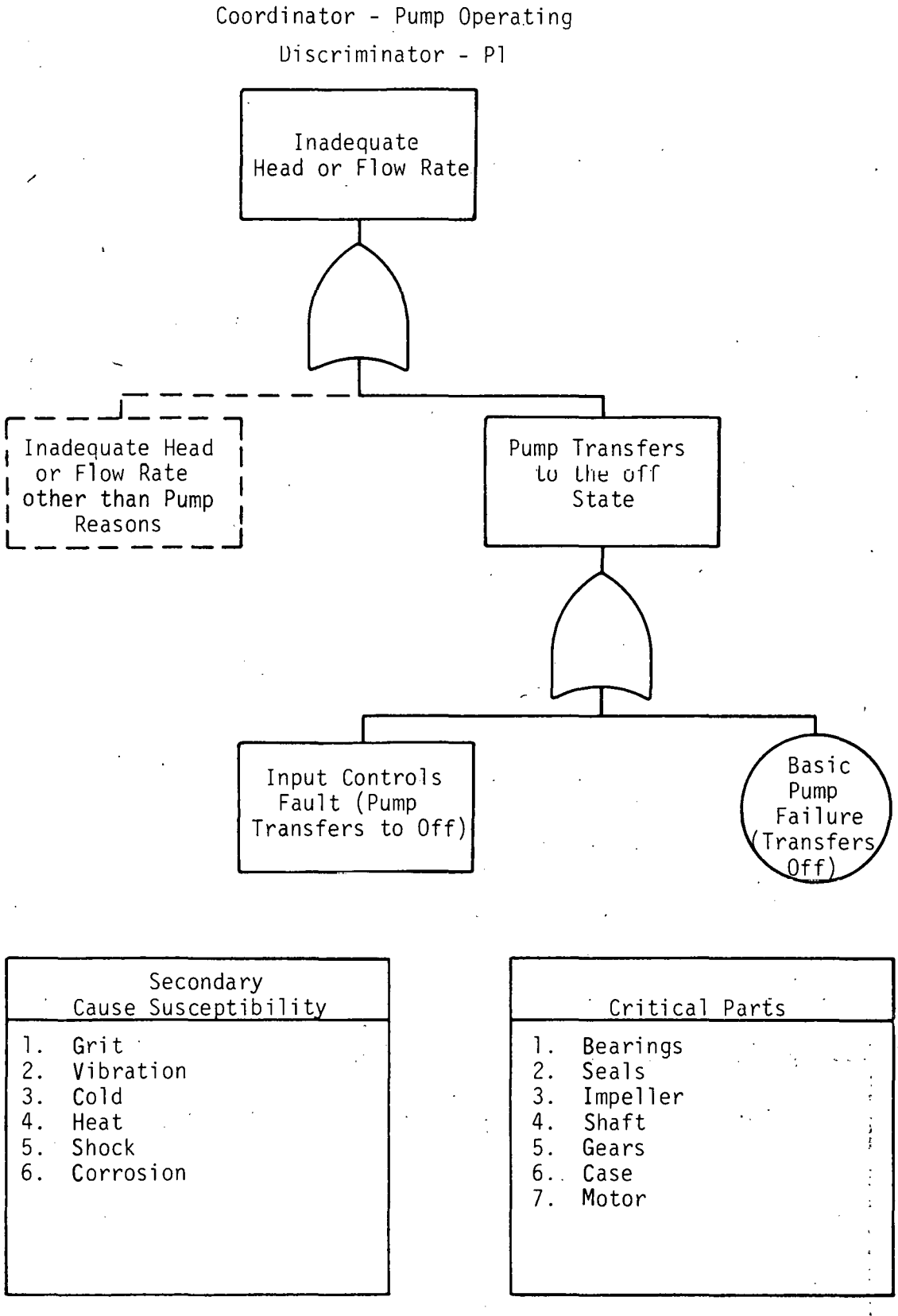
EXAMPLE

Mini Fault Trees

for

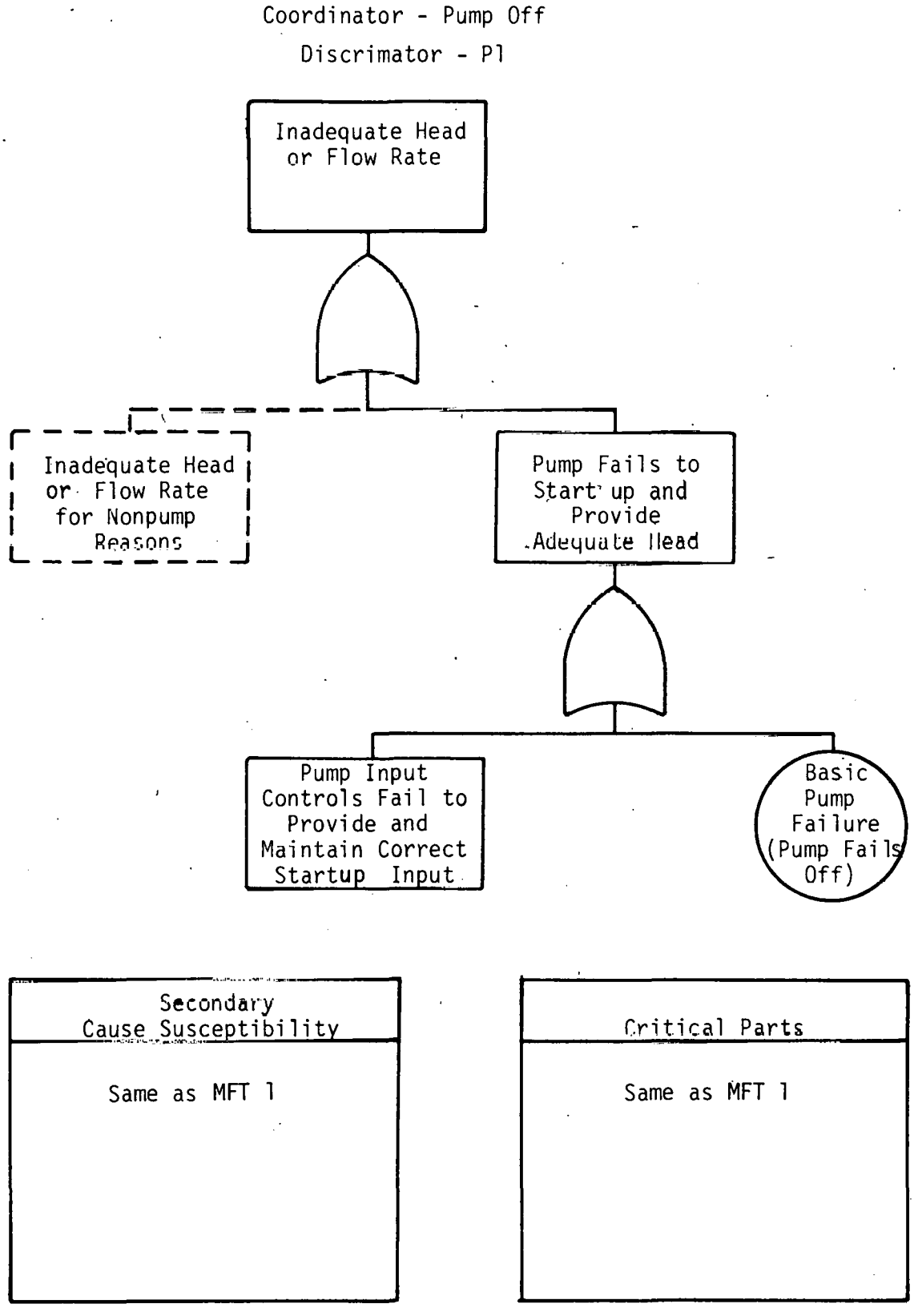
FREE SURFACE SODIUM PUMP

MFT Number	Output Event	Coordinator
1	Inadequate Head or Flow Rate	Pump operating
2	Inadequate Head or Flow Rate	Pump off
3	Excessive Head or Flow Rate	Pump operating or Pump off



[a] MFT 1 for free surface sodium pump.

Fig. 14 MFT for free surface sodium pump.

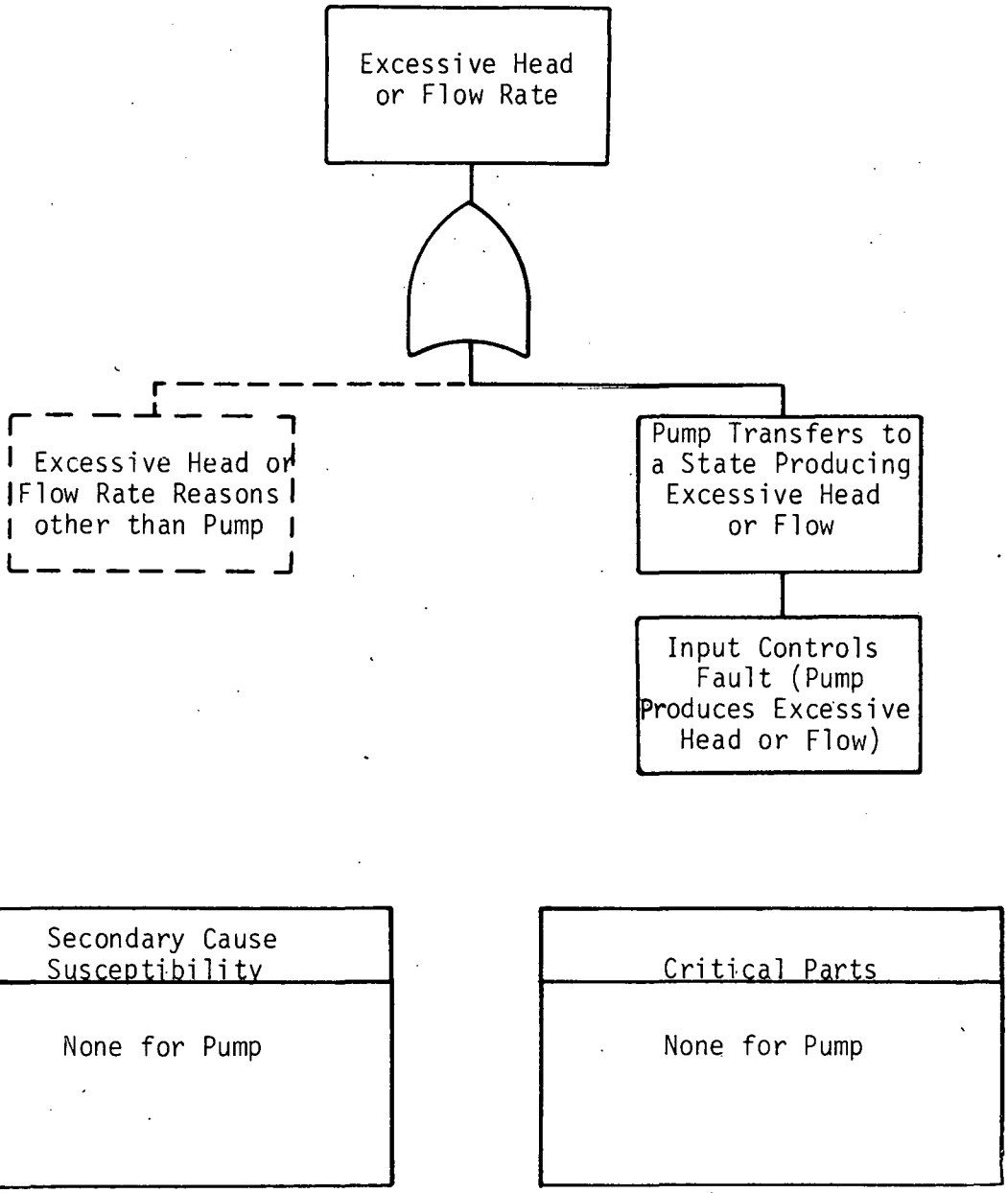


[b] MFT 2 for free surface sodium pump.

Fig. 14 MFT for free surface sodium pump (contd.).

Coordinator - Pump Operating or Pump Off

Discriminator - P2



[c] MFT 3 for free surface sodium pump.

Fig. 14 MFT for free surface sodium pump (contd.).

VI. CONSOLIDATING MFT LIBRARY INFORMATION

In practice, MFT information should be consolidated both within the MFT for a given component and among components. In the example given in Section V-4, the free surface sodium pump has two operating states and four output events giving rise to eight MFT. These eight MFT were consolidated into three representations.

Consolidation is also possible among components. The MFT for a sodium isolation valve were developed and are given in the following example and Figure 15. By selecting terminology appropriately, the MFT for this valve are seen to be appropriate for any sodium valve.

EXAMPLE

Mini Fault Trees

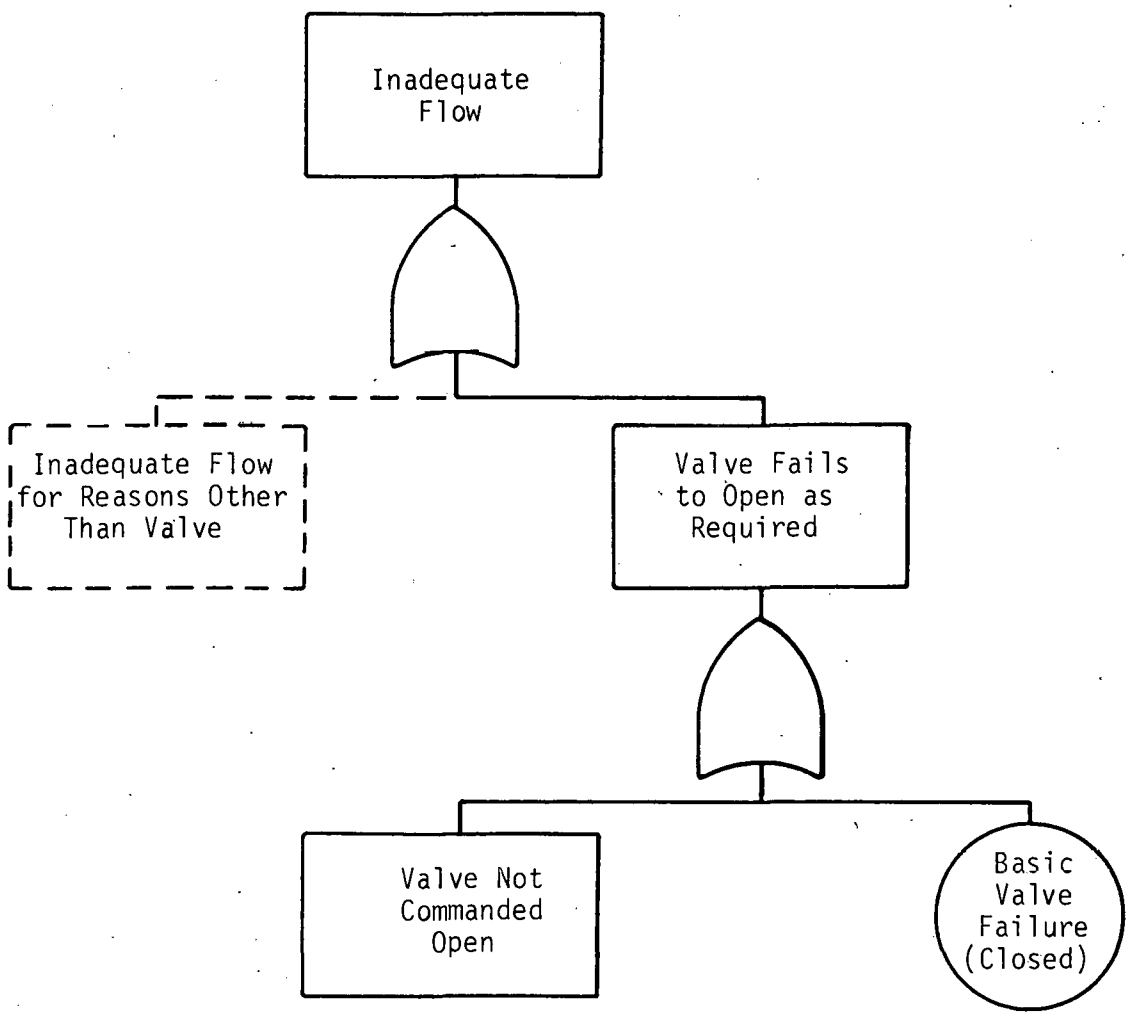
for

SODIUM VALVE

MFT Number	Output Event	Coordinator
1	Inadequate Flow	Valve closed
2	Inadequate Flow	Valve open
3	Inadvertent Flow	Valve closed
4	Inadvertent Flow	Valve open

Coordinator - Valve Closed

Discriminator - V1

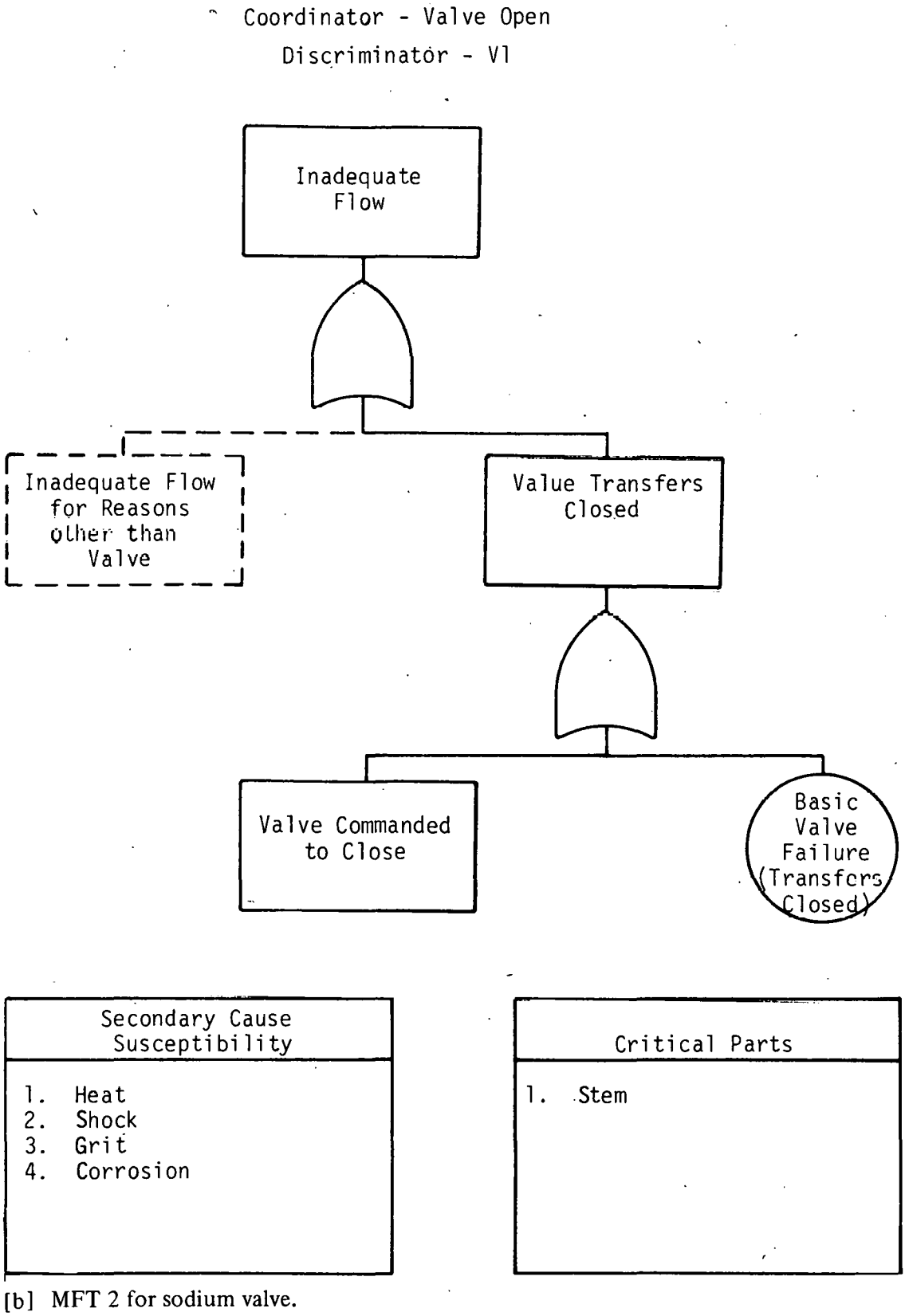


Secondary Cause Susceptibility
1. Heat
2. Grit
3. Shock
4. Vibration
5. Excessive Control Forces
6. Corrosion

Critical Parts
1. Seat and Gate Coatings
2. Stem (Shaft)
3. Seat
4. Gate
5. Packing
6. Sleeve

[a] MFT 1 for sodium valve.

Fig. 15 MFT for sodium valve.

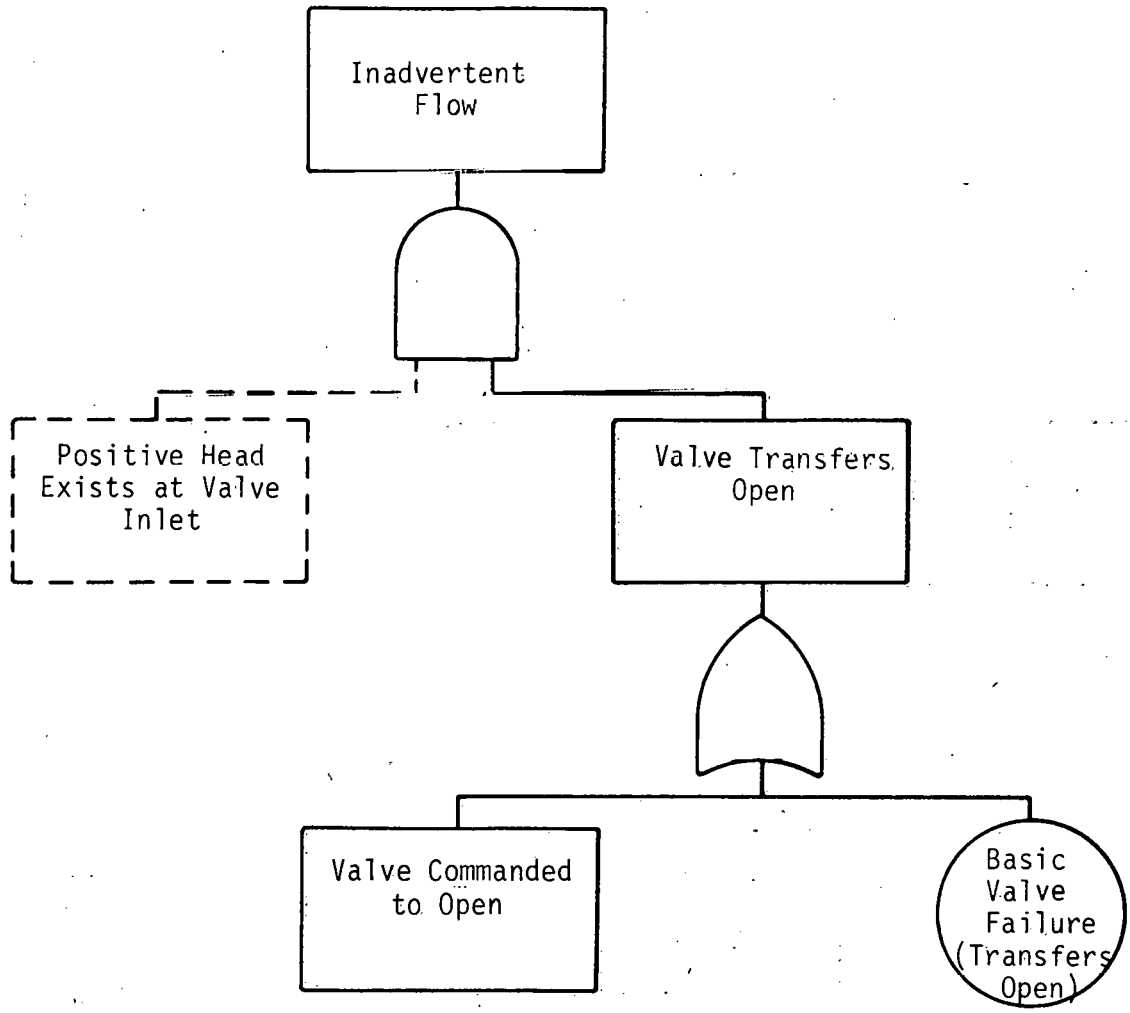


[b] MFT 2 for sodium valve.

Fig. 15 MFT for sodium valve (contd.).

Coordinator - Valve Closed

Discriminator - V2

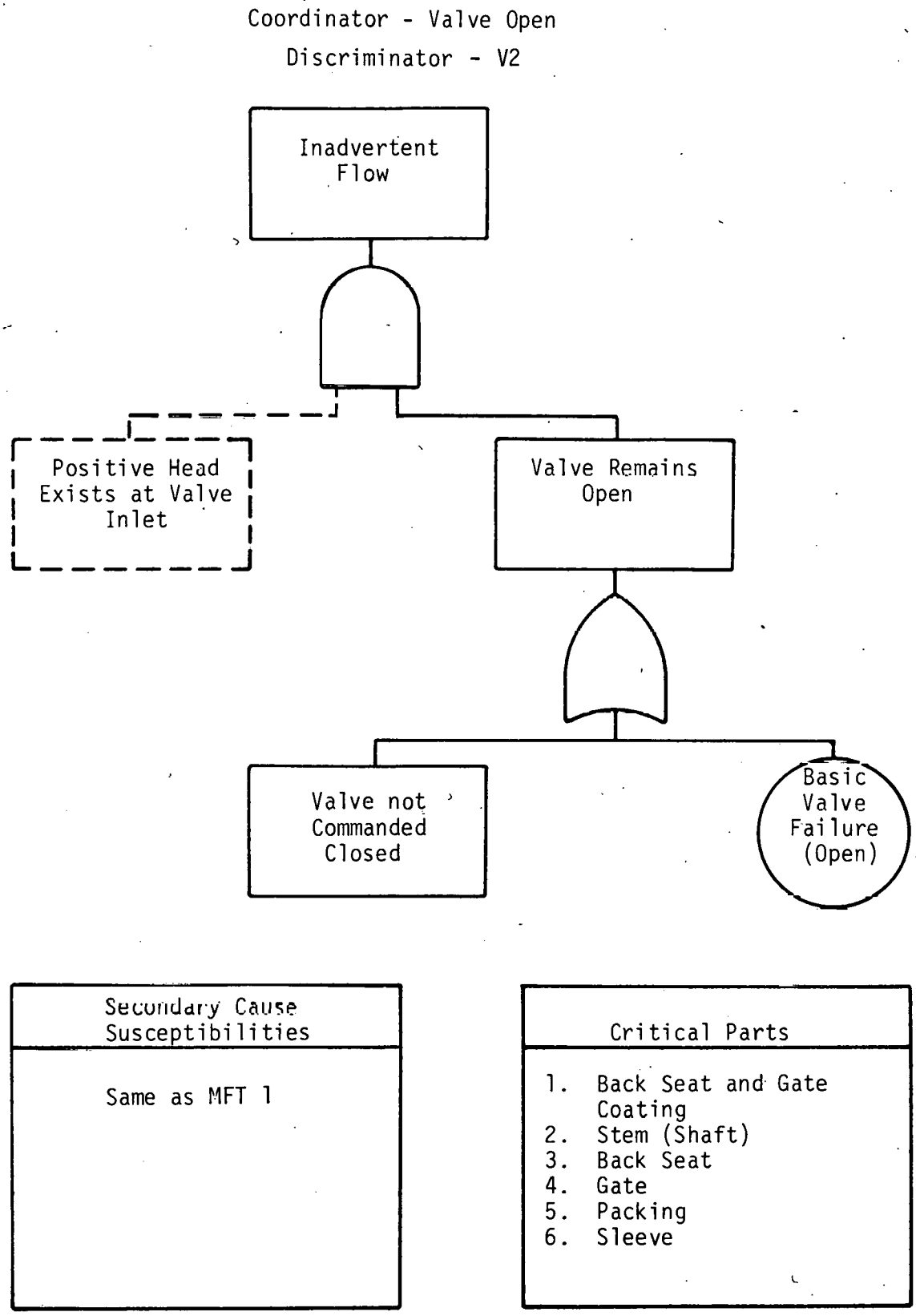


Secondary Cause Susceptibilities
Same as MFT 1

Critical Parts
1. Stem (Shaft)
2. Seat
3. Gate
4. Packing
5. Sleeve

[c] MFT 3 for sodium valve.

Fig. 15 MFT for sodium valve (contd.).



[d] MFT 4 for sodium valve.

Fig. 15 MFT for sodium valve (contd.).

VII. CONCLUSIONS AND RECOMMENDATIONS

A MFT library affords the analyst a mechanism for storage and retrieval of system independent component failure information. This library then has potential for reducing the effort required for logic model construction for the complex systems encountered in practice. In no way does the MFT library reduce the importance of understanding the system to be analyzed.

The MFT library offers a first step toward standardization of logic model construction. The library should be made available on a widespread basis and a mechanism established to accept contributions from all users.

The library initially can be in the form of looseleaf notebooks containing sets of MFT for various components cataloged in a manner compatible with failure data collection systems now being used. Eventually computer-aided storage and retrieval of MFT information should be implemented.

Two methods exist for obtaining information for the MFT library. The information can be actively sought in a program specifically designed to create the library, or, alternately, the MFT library can be evolved in conjunction with separately funded reliability and safety applications. This latter method requires an overseeing function for the library to ensure that sound and consistent format is utilized and that appropriate material is used in a concise manner. The latter method is less costly. The former method results in a library in less amount of calendar time.

The disadvantages of a MFT library are:

(1) Initial cost of collecting the information

(2) Necessity for training of personnel to use the library

(3) Necessity of obtaining contributions for the library from users

(4) Cost of maintaining the library.

The advantages of the library include:

(1) Reduced cost of reliability and safety analysis

(2) Reduced calendar time required for analyses

(3) Ability to utilize analysis personnel with less experience with the system components

(4) Reducing the routine associated with reformulating failure logic of complex components.

At this point the advantages appear to far outweigh the disadvantages.

VIII. REFERENCES

1. *Reporting Procedures Manual for Nuclear Plant Reliability Data System,* Southwest Research Institute, San Antonio, Texas (July 1973).

2. J. B. Fussell, *Synthetic Tree Model: A Formal Methodology for Fault Tree Construction,* ANCR-1098 (March 1973).

3. G. Powers, *Computer Aided Synthesis of Fault Trees,* NATO Advanced Study Institute on Generic Techniques of System Reliability Assessment, Nordhoff Publishing Company (1974).

4. D. S. Nielsen, *The Cause Consequence Diagram Method as a Basis for Quantitative Accident Analysis,* RISØ-M-1374 (May 1971)[a].

5. J. R. Taylor, *Sequential Effects in Failure Mode Analysis,* RISØ-M-1740 (August 1974)[a].

6. *Failure Data Handbook for Nuclear Power Facilities, Vol. I, Failure Data and Applications Technology,* Liquid Metal Engineering Center, LMEC-Memo-69-7, Atomics International (May 1975).

7. *Sodium Technology,* Atomics International, North American Rockwell, Canoga Park, Calif. (1970).

---

[a] Available from the Library of the Danish Atomic Energy Commission, RISØ, DK-9000 Roskilde, Denmark.

REPORT IV

THE IMPLEMENTATION OF PHASED MISSION TECHNIQUES TO NUCLEAR SYSTEMS ANALYSIS

G. R. Burdick

J. B. Fussell

D. M. Rasmuson

J. R. Wilson

ABSTRACT

This report introduces an approach for implementation of a methodology for phased mission analysis. Phased missions and related concepts are defined and illustrated using simple examples. Approximation methods for calculating mission unreliability and unavailability are discussed. A boiling water reactor phased mission example is presented in detail.

ACKNOWLEDGMENT

We acknowledge the contributions of Professor J. D. Esary of the Naval Postgraduate School and CDR H. Ziehms (F.R.G.) to phased missions analysis and express our gratitude to them and to LCDR M. G. Bell (USN) for their generous advice and assistance.

CONTENTS FOR REPORT IV

FIGURES

TABLE

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

THE IMPLEMENTATION OF PHASED MISSION TECHNIQUES TO NUCLEAR SYSTEMS ANALYSIS

I. INTRODUCTION

One of the most important but least understood problems in systems unreliability analysis has been the phased mission problem. A phased mission is a task, to be performed by a system, during the execution of which the system is altered such that the logic model changes at specified times. Thus, during a phased mission there are time periods (phases) during which either the system configuration, system failure characteristics, or both are distinct from those of any immediately succeeding phase. A most important phased mission problem is to calculate or obtain bounds for mission unreliability, where mission unreliability is defined as the probability that the system fails to function successfully through all of the phases.

There are different types of phased missions, each giving rise to its own phased mission problems. The components which comprise the system may fail independently of each other or have interdependent failure properties. The components may be reparable, with specified repair times, or they may be nonreparable. Often a system undergoing a phased mission will contain both reparable and nonreparable components. In a mission such as that of an intercontinental ballistic missile, all of the components are considered nonreparable. During a manned space flight, however, it may be possible for an astronaut to replace or at least repair a malfunctioning item. This latter situation has in fact occurred. A very elegant treatment of the case where all of the components are nonreparable and have independent failure characteristics was recently provided by H. Ziehms[1] and J. D. Esary and H. Ziehms[2]. The main objectives of this report are to present their method of approach and to apply their methodology to a typical phased mission problem which might arise in the nuclear power industry. References 3, 4, and 5 give approaches taken by other investigators.

The need for application of phased mission analysis techniques to nuclear reactor systems is evident. A standard assumption has been that if a safety system is available when needed, it will function as long as required. Therefore, the system asymptotic unavailability became the reliability factor of merit. The actual problem is more difficult. The meaningful factor of merit is the system "dependability", that is, the unconditional probability of mission success. The concept of dependability is often, and somewhat loosely, illustrated as the product of system availability and reliability. Analysis of a hypothetical reactor accident chain is yet an even more complex problem since the "unreliability" required for the dependability calculation is obtained only through detailed application of phased mission techniques.

Before the results of Esary and Ziehms are described in detail, two examples used by them[2] are presented to illustrate some features of the phased mission problem they considered. Here, and throughout this report, all components of a system will be considered

nonreplaceable and nonreparable. Each component in the system is assumed to be functioning at the start of the first phase and to do so continuously in time until failure occurs. Once failure does occur, the component remains failed. Components displaying this type of behavior are said to "have a life"[6].

The first example illustrates a logical error committed by many phased mission investigators who supposed that correct mission unreliability could be found by simply calculating the probabilities of failure of individual phases and then forming the sum of these. The major flaw in this approach is the fact that every component is assumed to be functioning at the start of each phase.

Example 1: A system with two independent components, C_1 and C_2, is designed for a two phased mission. In order for the system to perform the required tasks, at least one component has to function through the first phase and both components have to function through the second phase. The fault tree for this system is as shown in Figure 1.

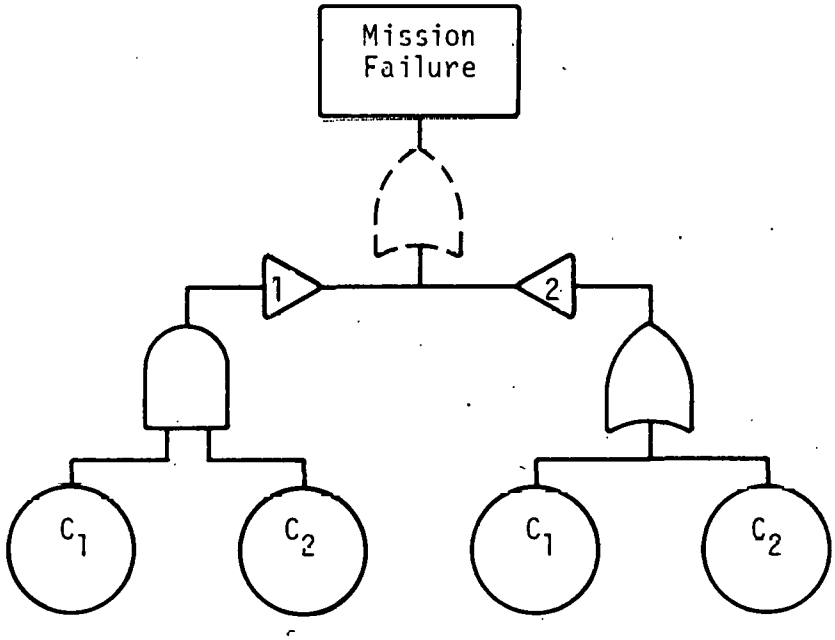


Fig. 1 Example 1 fault tree.

For k = 1, 2, the probability that component C_k functions through Phase 1, is denoted by π_{k1} and the conditional probability that component C_k functions through Phase 2 is denoted by π_{k2} given that it has functioned through Phase 1. The system reliability for Phase 1 is $\pi_1 = \pi_{11} + \pi_{21} - \pi_{11}\pi_{21}$, and the system reliability for Phase 2, given that both components have functioned through Phase 1, is $\pi_2 = \pi_{12}\pi_{22}$. The overall mission unreliability is then calculated as follows:

$$\bar{\pi} = 1 - \pi_1\pi_2 = 1 - (\pi_{11} + \pi_{21} - \pi_{11}\pi_{21})\,\pi_{12}\pi_{22} \cdot \tag{1}$$

Which is less than the correct mission unreliability, which is

$$\bar{p} = 1 - \pi_{11}\pi_{12}\pi_{21}\pi_{22} \tag{2}$$

because mission success is achieved if, and only if, both components function through both phases.

Although each of the components of a system has a life, the system itself may fail to have a life if the mission consists of more than one phase. Esary and Marshall[6] have shown that for a single phase mission, describable by a coherent[7] fault tree, it is sufficient that each component has a life to guarantee that the system has a life. The second example illustrates that component life does not imply system life for a multiphase mission.

Example 2: A two component system is designed for a two phase mission with the fault tree[a] given in Figure 2.

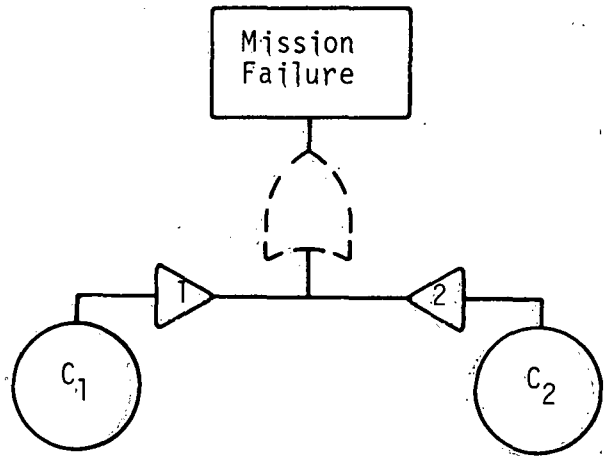


Fig. 2 Example 2 fault tree.

If π_{kj}, k = 1, 2, j = 1, 2, are defined as in Example 1, then there is a probability $(1 - \pi_{11})\pi_{21}\pi_{22}$ that the system fails in Phase 1, but functions again in Phase 2. In this sense the system does not have a life.

In light of this example system life must be defined apart from component life for a multiphase mission, that is, the life of a system will continue until the first unsuccessful phase occurs, at which time system life ends. Thus, end of system life and mission failure are coincident events.

---

[a] References 8, 9, and 10 give a complete discussion of fault trees.

The discussion of the phased mission technique of Esary and Ziehms, Section II, culminates in a tutorial example in which the major difficulty with the Esary-Ziehms method, the increase in the number of minimal cut sets, is made apparent. Included with the tutorial example is a discussion of time-dependent failure probabilities and an explanation of how to calculate the conditional probabilities which are vital to the Esary-Ziehms exact solution method. In addition to the tutorial example, an example from a boiling water reactor (BWR) situation is presented (Section IV). Exact solutions for mission unreliability for these examples are obtained using the Esary-Ziehms method and the MOCUS[11], PREP[12], and KITT[12] computer algorithms developed at the Aerojet Nuclear Company. These exact solutions are compared with approximate values obtained by methods discussed in Section III.

II. THE PHASED MISSION METHOD OF ESARY AND ZIEHMS

The Esary-Ziehms technique was derived assuming that (a) all components in the system are nonreparable and nonreplaceable and (b) all components have independent failure characteristics. Although argument may be presented that assumption (b) is rarely met in practice it is also true that, to date, except for very trivial systems, no general method of solution is known for the case where the components have interdependent failure characteristics. Thus assumption (b) or other assumptions must be made to obtain bounds on system unreliability. Common cause and secondary failure analysis[a] are two areas where interdependent failure characteristics are now being investigated.

The method of Esary and Ziehms combines the techniques of cut set cancellation and a component transformation which reduces the original multiphase mission into an equivalent single phase mission. Under the transformation, the phase configurations of the system become subsystems which act in series. These techniques are explained in detail through further use of examples provided by Esary and Ziehms[4].

1. CUT SET CANCELLATION[b]

By definition, a cut set is a collection of component failure modes such that if all of them occur then the system fails. A minimal cut set is a cut set which does not properly contain any other cut set. For brevity "component" will be used for "component failure mode" when the discussion concerns fault tree events.

For Example 1 of Section I, the minimal cut set in Phase 1 is $\{C_1, C_2\}$. In Phase 2 the cut sets are $\{C_1\}$, $\{C_2\}$. $\{C_1, C_2\}$ contains a cut set (in fact, both cut sets) in Phase 2; hence, over the mission, $\{C_1, C_2\}$ is not minimal. To consider $\{C_1, C_2\}$ as a set of components causing mission failure is redundant, because if either C_1 or C_2 fail in Phase 1 the mission will not succeed because both components are necessary for the success of Phase 2. Cancellation of $\{C_1, C_2\}$ in Phase 1 leaves the mission fault tree shown in Figure 3.

Although Figure 3 is a simpler fault tree for the mission of Example 1, the question of what probabilities of failure to assign to the components C_1 and C_2 still remains; that is, even though $\{C_1, C_2\}$ was "canceled" in Phase 1, the fact that either C_1 or C_2 could have failed in Phase 1 as well as Phase 2 must be recognized. The factors on the right hand side of Equation (2), $\rho = 1 - \pi_{11}\pi_{12}\pi_{21}\pi_{22}$, suggest that further fault tree modification is necessary. By replacing C_1 with new components C_{11}, C_{12}, which act in series, and

---

[a] Reports I and II in this document.

[b] References 3,4, and 5 present further discussion of this subject as applied to phased missions.

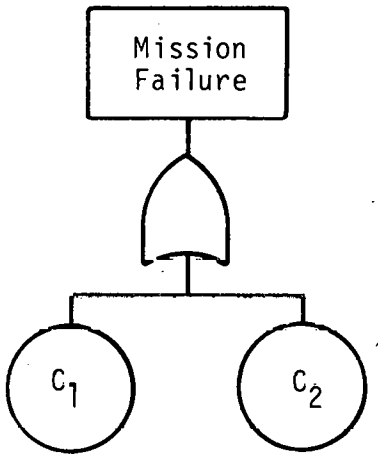


Fig. 3 Example 1 fault tree after cut cancellation.

replacing C_2 with new components, C_{21}, C_{22} acting in series, the fault tree shown in Figure 4 is obtained. The probabilities of failure assigned to component C_{ij}; i = 1, 2; j = 1, 2 are simply $\bar{\pi}_{ij} = 1 - \pi_{ij}$; i = 1, 2; j = 1, 2. The end result is thus an equivalent single phase mission fault tree where the probability of failure agrees with that of the original mission of Example 1.

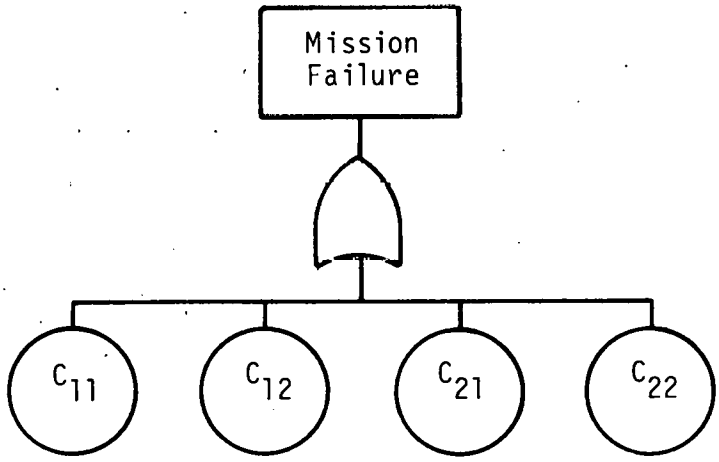


Fig. 4 Example 1 fault tree after cut cancellation and component transformation.

To further illustrate the subject of cut cancellation; the fault tree in Figure 5 is considered. Here the cut set $\{C_1, C_2\}$ cannot be ignored for the mission because if both C_1 and C_2 survive Phase 1, they could both fail in Phase 2. Before the rule for cut set cancellation is stated, a slightly more complex example is presented to illustrate the technique. The example is again due to Esary and Ziehms[2] but is modified to fit the fault tree convention.

Example 3: A mission has the fault tree shown in Figure 6.

The minimal cut sets are:

Phase 1 $\{C_1\}$ $\{C_2, C_3\}$

Phase 2 $\{C_2\}$ $\{C_1, C_3\}$

The Phase 1 cut set $\{C_2, C_3\}$ contains the Phase 2 cut set $\{C_2\}$, and so can be canceled in Phase 1. No cancellation results from the fact that the Phase 2 cut set $\{C_1, C_3\}$ contains the Phase 1 cut set $\{C_1\}$. After cut cancellation, the fault tree is simplified to that shown in Figure 7.

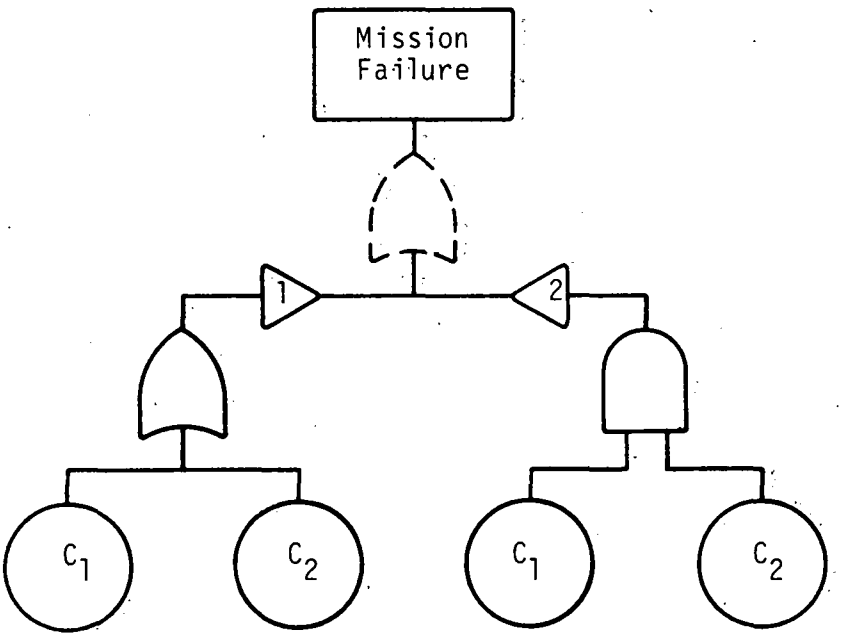


Fig. 5 Cut cancellation counterexample for Example 1.

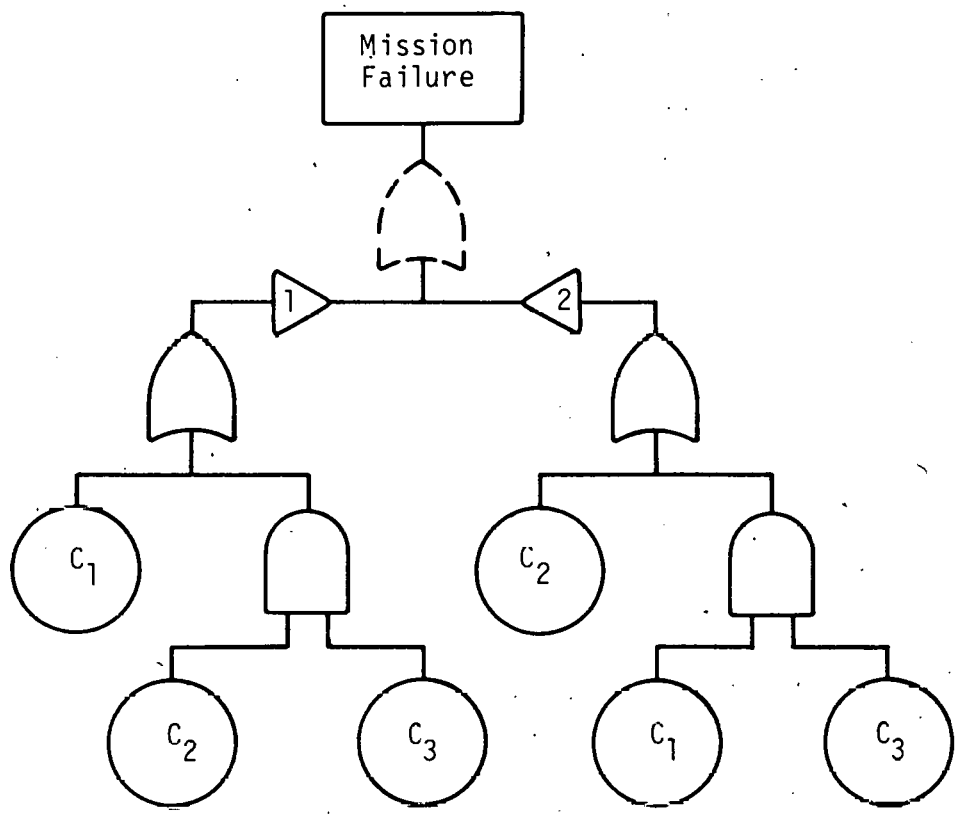


Fig. 6 Example 3 fault tree.

The method of cut cancellation can be summed up in the following general rule:

> "A minimal cut set in a phase can be canceled, i.e., omitted from the list of minimal cut sets for that phase, if it contains a minimal cut set of a later phase."[2]

Similarly, and for emphasis, if a minimal cut set in a phase contains a minimal cut set in an earlier phase, cut cancellation is not justified at that stage of the analysis.

2. COMPONENT TRANSFORMATION

In the previous section, the fault tree of Figure 3 was replaced by that of Figure 4 because the latter presented a more direct way to obtain the correct mission unreliability equation, Equation (2). Before the cut set cancellation (which changes Figure 1 to Figure 3) is performed, the component C_1 is replaced by C_{11}, C_{12} and C_2 is replaced by C_{21}, C_{22}. The result may be viewed as a single phase mission for a system composed of subsystems operating in series; however, the resulting single phase mission is not equivalent to the original as Example 4 and Figure 8 illustrate.

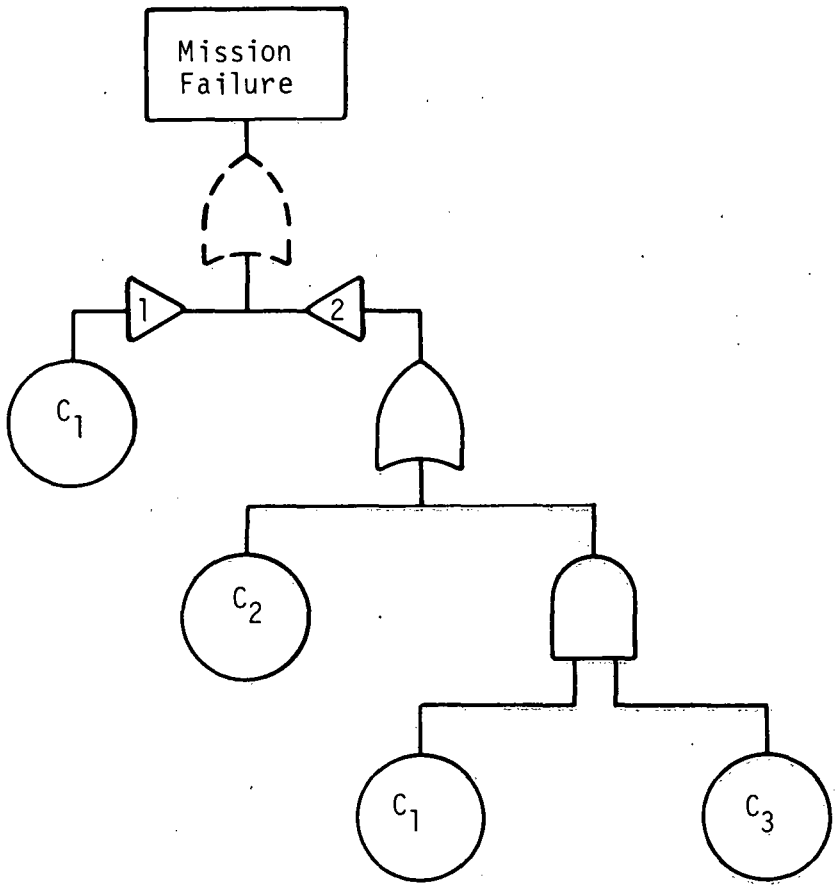


Fig. 7 Example 3 fault tree after cut cancellation.

Example 4: If π_{kj}, k = 1, 2; j = 1, 2 are defined as in Example 1, and $\rho_{k1} = \pi_{k1}$, $\rho_{k2} = \pi_{k1}\,\pi_{k2}$; k = 1, 2, the subsystem unreliabilities are

$$\bar{\rho}_1 = 1 - \pi_{11} - \pi_{21} + \pi_{11}\pi_{21} = 1 - \rho_{11} - \rho_{21} + \rho_{11}\rho_{21}$$

$$\bar{\rho}_2 = 1 - \pi_{11}\pi_{12}\pi_{21}\pi_{22} = 1 - \rho_{12}\rho_{22} \quad .$$

If subsystem independence is assumed, the mission unreliability is given by $\bar{\rho} = \bar{\rho}_1 + \bar{\rho}_2 - \bar{\rho}_1\,\bar{\rho}_2$. Except in trivial cases, this unreliability is larger than the true system unreliability $\bar{\rho} = \bar{p}_2$ which is given by Equation (2) of Example 1. In their paper, Esary and Ziehms[2] prove that $\bar{\pi} \leq \bar{p} \leq \bar{\rho}$ is always true, where $\bar{\pi}$ and $\bar{\rho}$ are mission unreliabilities calculated in the manners illustrated by Examples 1 and 4, respectively, and $\bar{p}$ is the exact unreliability calculated by using the conditional probabilities that the components fail

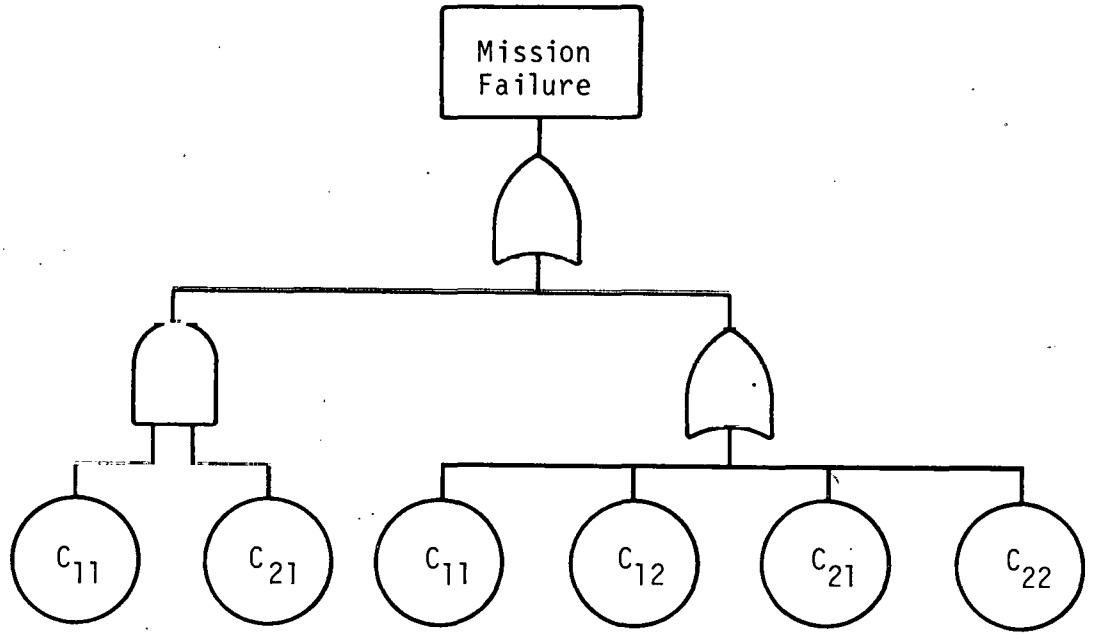
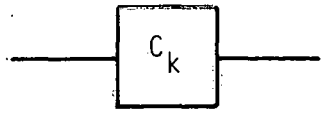


Fig. 8 Example 4 fault tree.

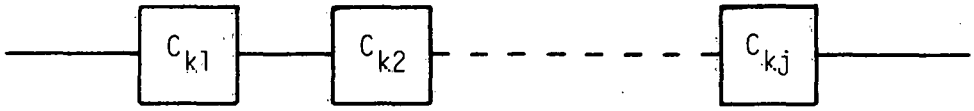
in a phase, given they have survived the previous phases, as the unreliabilities in the equivalent transformed system. In Example 4, the cut set $\{C_{11}, C_{21}\}$ may be canceled because it contains cut sets $\{C_{11}\}$, $\{C_{21}\}$ from Subsystem 2. This cut set cancellation, after the component transformation, results again in the fault tree of Figure 4. Esary and Ziehms[2] provide a proof that cut set cancellation does not affect the mission unreliability $\bar{p}$ when such cancellation is performed before component transformation. The performance of cut cancellation after component transformation also has no effect on exact mission unreliability. Before and after cut cancellations do, however, improve estimates of mission unreliability as explained in Section III. The component transformation technique is perhaps best summed up in the words of Esary and Ziehms[2]:

> "Complexities in the reliability analysis of phased missions arise because a component's performance in each phase depends on its performance in previous phases. The dependence, however, is of a special type. A component functions in phase j if, and only if, it has previously functioned in phase 1, and in phase 2, . . ., and in phase j-1, and then functions in phase j. This sequence of requirements suggests that the performance of a component in phase j can be represented by a series-like structure whose elements represent its performance in phases 1, . . .j.

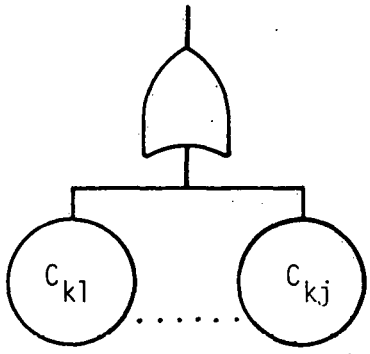
To be more specific, suppose that component C_k is replaced in phase j by a system of components $C_{k1}, \ldots, C_{kj}$, performing independently and in series. In block diagram format, the block



is replaced in phase j by



In fault tree format, the input event $\overline{C}_k$ (failure of component C_k) is replaced in phase j by



"

In summary, the complete reduction of the multiphase mission problem (in the nonreparable, independent failure case) into an equivalent single phase mission problem is accomplished by the following steps:

(a) Cut set cancellation is performed over the mission as described for Example 1 of Section II-1

(b) In the configuration for phase j, component C_k is replaced by a series system in which the components $C_{k1}, \ldots, C_{kj}$ perform independently with the probability of failure of C_{kj}, $\overline{\pi}_{kj}$, the conditional probability that component C_k fails in phase j given it has performed successfully in phases 1, 2, . . . , j-1

(c) The transformed phase configurations are considered to be subsystems operating in series in a new system involved in a single phase mission

(d) Cut cancellations are performed over the single phase mission.

3. A THREE PHASE TUTORIAL EXAMPLE

Presented here is an example due to Esary and Ziehms[2] which has been modified to follow the fault tree format.

Example 5: A fire department has three vehicles:

A multipurpose fire engine (M)

A tanker (T)

A light fire truck (L).

The firefighting equipment of a small chemical factory located nearby consists of:

A sprinkler system (S)

A hydrant (H)

A special apparatus for fighting chemical fires (F).

The plant safety engineer wonders whether the combined hardware resources of the fire department and the factory are sufficient to fight a fire in the factory. He consults the fire chief, and together they conclude:

(1) During the initial stage of a fire either the multipurpose engine, which carries a small water supply, or the light truck and the sprinkler system suffices to evacuate the building.

(2) To contain the fire, the special apparatus in the factory is needed, together with some auxiliary capability from the multipurpose engine or the light truck. Water can be supplied to the special apparatus and the department's units by the hydrant, or if the hydrant is out of order, by the tanker through pumps in the multipurpose engine.

(3) After the fire has been contained it can be controlled either by the special apparatus or the multipurpose engine. Again, water can be supplied by the hydrant or by the tanker together with the multipurpose engine.

The system has six components and has to perform a three phased mission. A fault tree for this mission is shown in Figure 9.

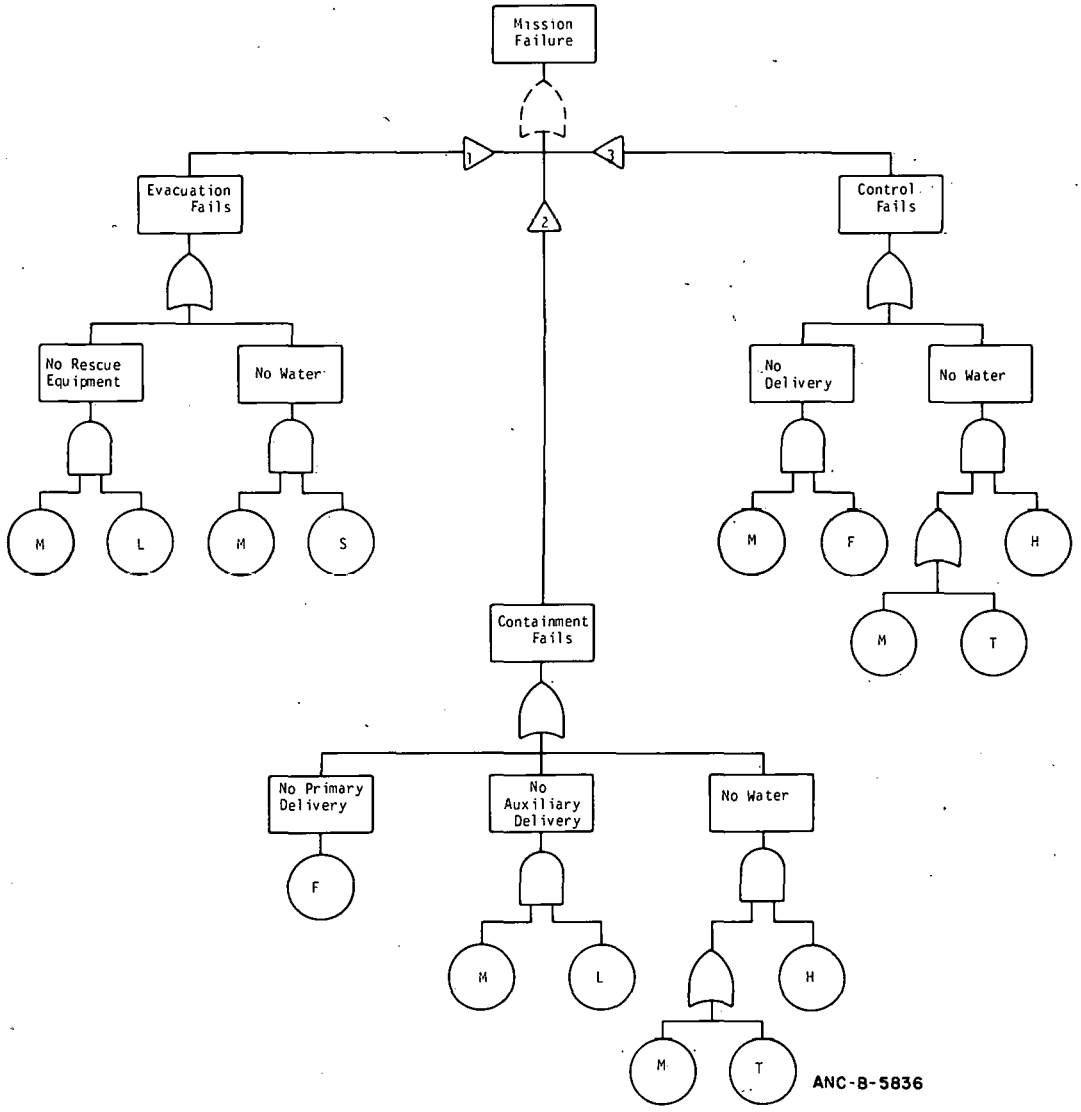


Fig. 9 Example 5 fault tree.

The minimal cut sets for this mission are, before cancellation:

Phase 1 {M, L} {M, S}

Phase 2 {F} {H, M} {H, T} {M, L}

Phase 3 {F, M} {H, M} {H, T}.

After cancellation [Step (a) of the previous section] the minimal cut sets are:

Phase 1 {M, S}

Phase 2 {F}{M, L}

Phase 3 {F, M}{H, M}{H, T}.

A fault tree for this simplified mission is shown in Figure 10.

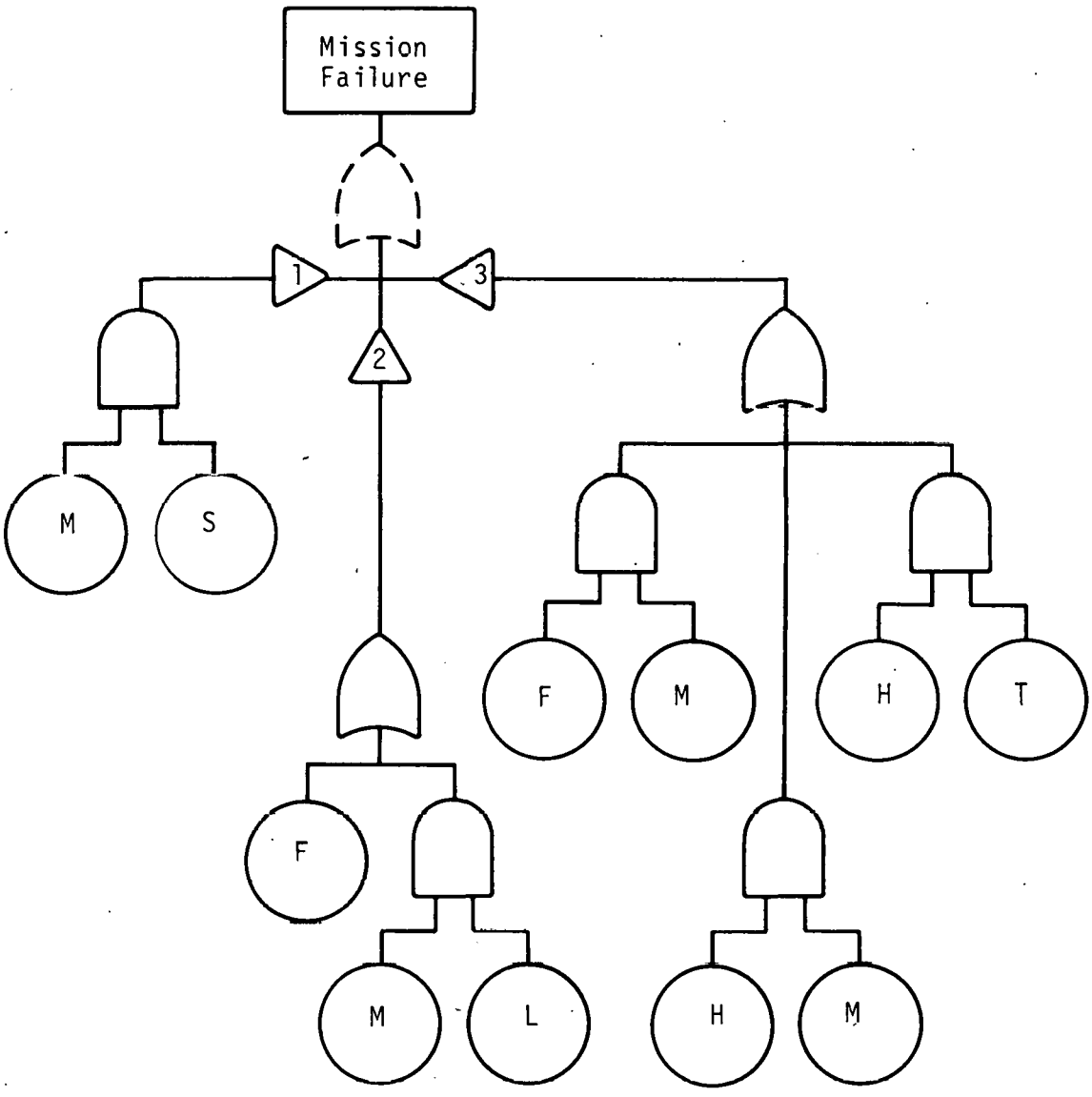


Fig. 10 Example 5 fault tree after cut cancellation.

Application of Steps (b) and (c) of the previous section to the three phase mission of Figure 10 results in the single phase mission of Figure 11. The transformed phases act as subsystems in series.

For this single phase mission the cut sets are:

Subsystem 1 $\{M_1, S_1\}$

Subsystem 2 $\{F_1\}$ $\{F_2\}$ $\{M_i, L_j\}$ i = 1, 2; j = 1, 2

Subsystem 3 $\{F_i, M_j\}$ $\{H_i, M_j\}$ $\{H_i, T_j\}$ i = 1, 2, 3,; j = 1, 2, 3.

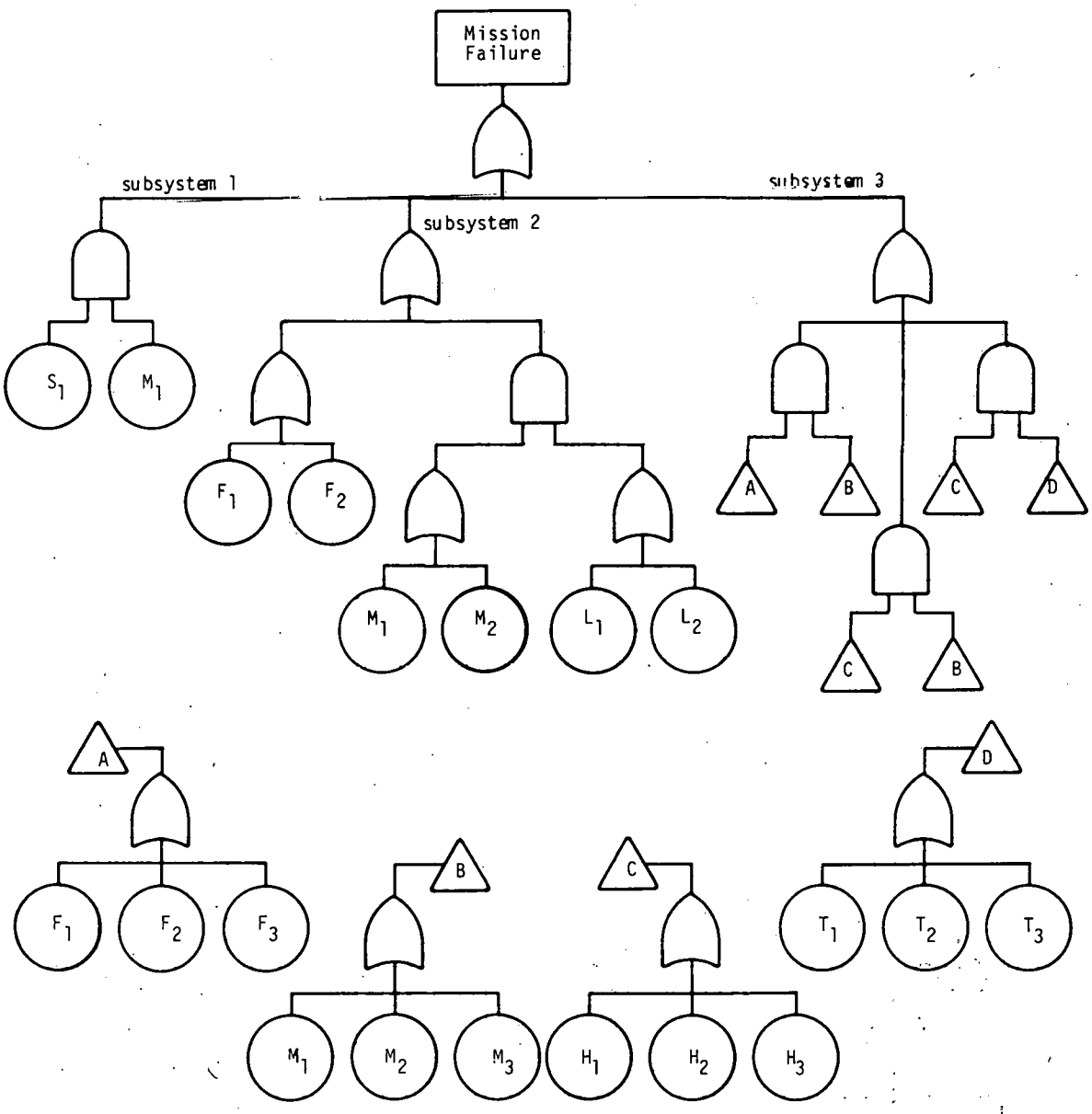


Fig. 11 Example 5 fault tree after cut cancellation and component transformation.

Now Step (d) is applied to this list of cut sets. The only sets which cancel are quickly seen to be those in Subsystem 3 which contain F_1 or F_2 from Subsystem 2. This reduces the list of cut sets (now minimal) to:

$\{M_1, S_1\}\{F_1\}\{F_2\}\{M_i, L_j\}\ i = 1, 2; j = 1, 2$

$\{F_3, M_j\}\{H_i, M_j\}\{H_i, T_j\}\ i = 1, 2, 3; j = 1, 2, 3.$

The equivalent single phase mission fault tree is shown in Figure 12.

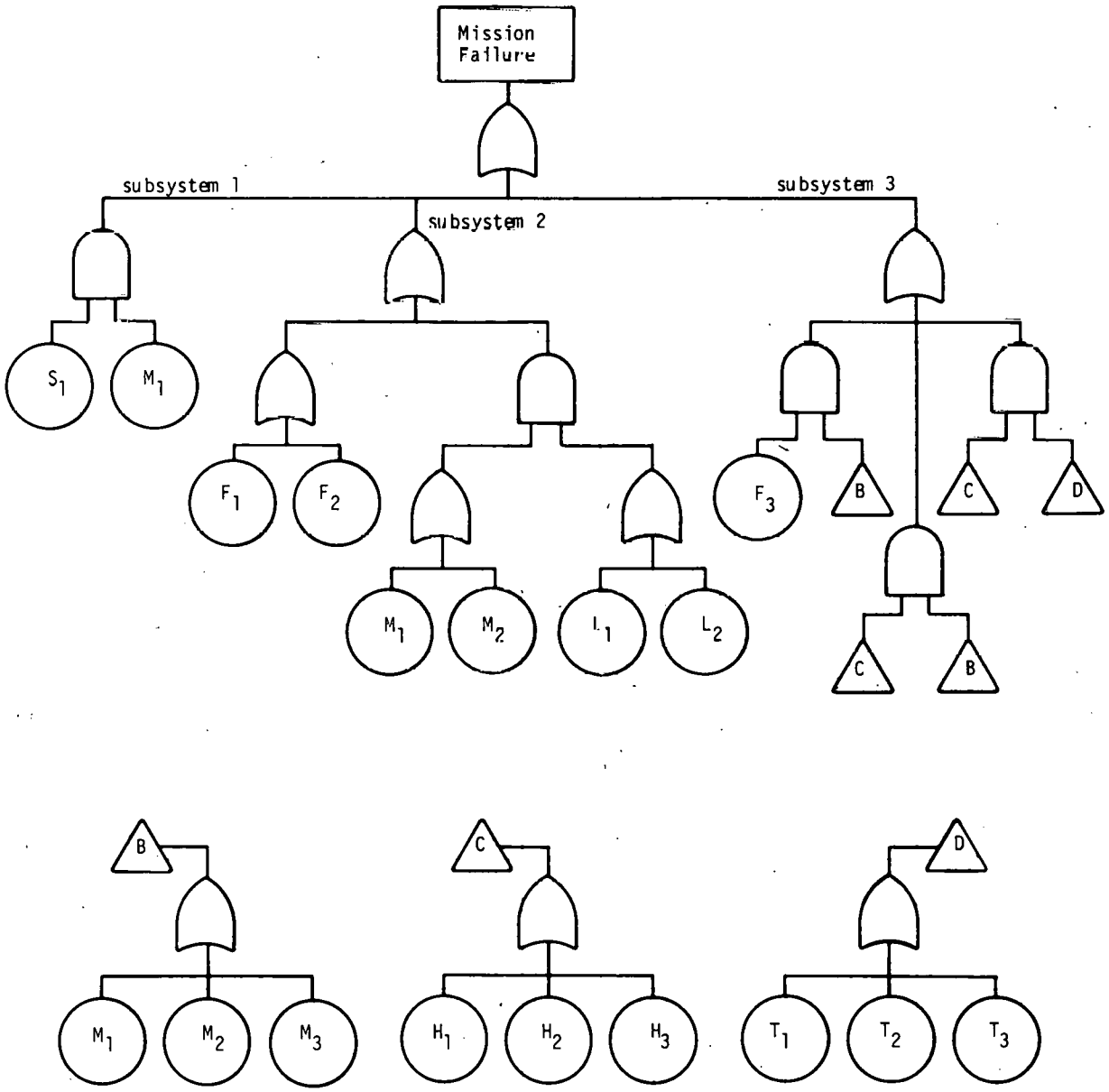


Fig. 12 Example 5 fault tree after application of Steps (a) through (d) of Section II.

The number of minimal cut sets has increased from 9 for the fault tree of Figure 9 to 28 for the fault tree of Figure 12. This increase in cut sets is an inherent problem with the component transformation of Esary and Ziehms, which could cause an exact solution, for a large system with several phases, to be quite difficult to obtain. If the system size and number of phases are large enough to cause calculational problems for an exact solution, the Esary and Ziehms approximation techniques may be employed. Four of these techniques are discussed in Section III and are compared with the exact solutions obtained for the problem of this section and for those of Section IV.

4. EXACT RELIABILITY CALCULATION

For the preceding example, the exact solution for mission unreliability may easily be hand calculated if the failure rates of the different pieces of equipment are known. Before these calculations are described, some preliminary information on probabilities is provided. Reference 13 provides additional information

A component in a three phase mission with constant failure rates λ_j, j = 1, 2, and 3 in phases I, II, and III, respectively, is considered. In Figure 13, the times at which the phases end is denoted by t_j, j = 1,2, and 3.

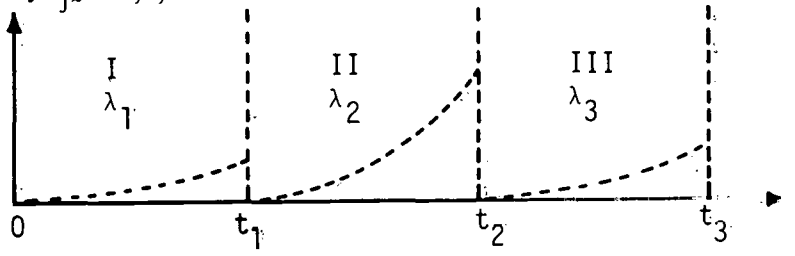


Fig. 13 Conditional component unreliability.

If the failure probability density function for the i^{th} component is exponential, then the probability of the i^{th} component having failed in the j^{th} phase is conditional on its having survived the previous phases and is given by

$$\pi_{ij} = \lambda_i \int_0^{t_j - t_{j-1}} e^{-\lambda_i \mu} \, d\mu = 1 - e^{-\lambda_i (t_j - t_{j-1})}, \quad j = 1, 2, 3. \tag{3}$$

Figure 13 is a graph for the case for which the λ's are distinct. For example purposes, the λ's for the various components in the tutorial example were taken to be equal over all the phases. The components and their assigned failure rates are listed in Table I. The end-of-phase times chosen were t_1 = 0.25 hr, t_2 = 1.25 hr, and t_3 = 11.25 hr. The conditional probabilities necessary for calculation of the exact Esary-Ziehms solution may be obtained by using the list of cut sets from Section II-3 step (d), the λ's from Table I, and

Equation (3), and by applying the inclusion-exclusion principle. By way of illustration, the two cut sets , $\{F_3, M_1\}$, $\{F_3, M_2\}$ are considered, and the probability of one or the other causing mission failure is calculated. That is,

$$P(\{F_3, M_1\} \cup \{F_3, M_2\}) = P\{F_3, M_1\} + P\{F_3, M_2\} - P(\{F_3, M_1\} \cap \{F_3, M_2\})$$

$$= P\{F_3\}\, P\{M_1\} + P\{F_3\}\, P\{M_2\} - P\{F_3\}\, P\{M_1\}\, P\{M_2\} \qquad (4)$$

$$= P\{F_3\}\,(P\{M_1\} + P\{M_2\} - P\{M_1\}\, P\{M_2\}) \quad .$$

TABLE I

FAILURE RATES FOR COMPONENTS OF EXAMPLE 5

H	Hydrant	$\lambda_H = 0.0001/hr$
F	Special Apparatus	$\lambda_F = 0.009/hr$
L	Light Truck	$\lambda_L = 0.003/hr$
M	Multipurpose Engine	$\lambda_M = 0.005/hr$
T	Tanker	$\lambda_T = 0.001/hr$
S	Sprinkler System	$\lambda_S = 0.0005/hr$

From Equation (3), and Table I:

$P\{F_3\} = 1 - \exp[-(0.009)\ 10]$

$P\{M_1\} = 1 - \exp[-(0.005)\ 0.25]$

$P\{M_2\} = 1 - \exp[-(0.005)\ 1]$.

Substitution of these values into Equation (4) then results in $P(\{F_3, M_1\} \cup \{F_3, M_2\}) = 5.36253 \times 10^{-4}$. In an analogous manner, the mission failure probability can also be calculated using all of the cut sets. The overall mission unreliability calculated for this example, using the KITT-1 computer program, was 1.59306×10^{-2}.

III. APPROXIMATION TECHNIQUES FOR MISSION UNRELIABILITY

So far, this report has been concerned with the calculation of mission unreliability for a phased mission. Unreliability, at time t, is the probability that the system has experienced its first failure prior to t. Closely connected to the concept of unreliability is that of unavailability. Unavailability is the probability that the system is not functioning at a particular instant in time, irrespective of failure it has experienced prior to that time.

In this section approximation methods for the mission unreliability are discussed. One approximation method employs the use of unavailability. These approximation methods are then illustrated using the tutorial example.

1. UNRELIABILITY APPROXIMATIONS

Esary and Ziehms developed several methods for obtaining conservative bounds on mission unreliability. Four of these methods were chosen for discussion here.

The following quantities will be used in the discussion of the unreliability bounds. The number $\overline{R}^*$ is the value obtained from applying the inclusion-exclusion principle[1] to a specified collection of cut sets.

The number $\overline{r}^*$ will denote the probability of failure of a cut set. Hence, if $A_1 \ldots A_k$ are components in a cut set, then $\overline{r}^*$ for this cut set is given by

$$\overline{r}^* = \prod_{\ell=1}^{k} P(\overline{A}_\ell) \tag{5}$$

where $P(\overline{A}_\ell)$ denotes the probability of failure of component A_ℓ. In the following, the quantity $\overline{r}^*$ will be subscripted as $\overline{r}^*_{ij}$ where i is the phase and j is the cut set within that phase.

1.1 Method $\overline{\rho}_{PRF}$

The $\overline{\rho}_{PRF}$ method consists of the following steps:

(1) The minimal cut sets for each phase are obtained from the appropriate logic model.

(2) $\overline{R}^*_i$ is calculated for the i^{th} phase using the appropriate cut sets and unconditional component unreliabilities.

(3) $\bar{\rho}_{PRF}$ is calculated by

$$\bar{\rho}_{PRF} = 1 - \prod_{i=1}^{m} (1-\bar{R}^*_i)$$

where m is the total number of phases in the mission. The number $\bar{R}^*_i$ of the i^{th} phase is in fact the system unavailability at the end of that phase.

Calculation of the exact unreliability of a mission is usually very laborious even by computer. It is usually approximated by the minimal cut upper bound[14], or even by the first terms in the expansion of the product. For example, if m = 3 in Step (3), then

$$\begin{aligned}
\bar{\rho}_{PRF} &= 1 - (1 - \bar{R}^*_1)(1 - \bar{R}^*_2)(1 - \bar{R}^*_3) \\
&= 1 - (1 - R^*_1 - \bar{R}^*_2 - \bar{R}^*_3 + \bar{R}^*_1\bar{R}^*_2 + \bar{R}^*_1\bar{R}^*_3 + \bar{R}^*_2\bar{R}^*_3 - \bar{R}^*_1\bar{R}^*_2\bar{R}^*_3) \\
&= \bar{R}^*_1 + \bar{R}^*_2 + \bar{R}^*_3 - \bar{R}^*_1\bar{R}^*_2 - \bar{R}^*_1\bar{R}^*_3 - \bar{R}^*_2\bar{R}^*_3 + \bar{R}^*_1\bar{R}^*_2\bar{R}^*_3 \\
&\leq \bar{R}^*_1 + \bar{R}^*_2 + R^*_3.
\end{aligned}$$

Another conservative approximation technique that applys to all the methods discussed is the approximation of the probability of failure by λt if $\lambda t < 0.1$[9] and ifan exponential distribution can be used to characterize the failure of components.

1.2 Method $\bar{\rho}_{PRF\text{-}CC}$

This method is similar to the $\bar{\rho}_{PRF}$ method just described. The difference that cut set cancellation between phases [Step (a) of Section II] is done before $\bar{R}^*_i$ is calculated for each phase. The numbers $\bar{R}^*_i$ calculated for each phase for this bound will in general be less than the $\bar{R}^*_i$ calculated in Step (2) of the previous method.

1.3 Method $\bar{\rho}_{PLB}$

This method consists of the following steps:

(1) Minimal cut sets are obtained for each phase from the appropriate logic model.

(2) $\bar{r}^*_{ij}$ is calculated for the j^{th} cut set in phase i using unconditional probabilities for each component.

(3) $\bar{R}^*_i$ is estimated for the i^{th} phase using the minimal cut upper bound; that is:

$$\bar{R}^*_i \approx 1 - \prod_{j=1}^{n_i} (1 - \bar{r}^*_{ij}) \qquad (6)$$

where n_i is the number of minimal cut sets in the i^{th} phase.

(4) $\overline{\rho}_{PLB}$ is calculated by

$$\overline{\rho}_{PLB} = 1 - \prod_{i=1}^{m} (1-\overline{R}^*_i) \tag{7}$$

where m denotes the number of phases in the mission.

1.4 Method $\overline{\rho}_{PLB-CC}$

This method is identical to the $\overline{\rho}_{PLB}$ method except cut set cancellation [Step (a) of Section II-2] is performed before $\overline{R}^*_i$ is calculated for each phase [Step (3)].

1.5 Remarks

Ziehms[1] shows that the following ordering exists among the bounds:

$$\overline{p} \leq \overline{\rho}_{PRF-CC} \leq \left\{ \begin{array}{l} \overline{\rho}_{PLB-CC} \\ \\ \overline{\rho}_{PRF} \end{array} \right\} \leq \overline{\rho}_{PLB} \quad . \tag{8}$$

Here, $\overline{p}$ is the exact mission unreliability. No comparison can be made between $\overline{\rho}_{PRF}$ and $\overline{\rho}_{PLB-CC}$. The difference between $\overline{\rho}_{PRF-CC}$ and $\overline{\rho}_{PLB-CC}$ (and also $\overline{\rho}_{PLB}$ and $\overline{\rho}_{PRF}$) is in the calculation of the phase unreliability. In practice, either $\overline{\rho}_{PRF-CC}$ or $\overline{\rho}_{PLB-CC}$ may be used to estimate mission unreliability.

One approximation of $\overline{\rho}_{PRF-CC}$ or $\overline{\rho}_{PLB-CC}$ which is very useful for hand calculations is the following:

(1) $\overline{R}^*_i$ of a phase is estimated by summation of the $\overline{r}^*_{ij}$ of the cut sets

(2) The mission unreliability is estimated by summation over the $\overline{R}^*_i$ of the phases.

Thus

$$\overline{\rho}_{PRF-CC} \approx \sum_{L=1}^{m} \sum_{j=1}^{n_i} \overline{r}^*_{ij} \approx \overline{\rho}_{PLB-CC} \quad . \tag{9}$$

The λt approximation could also be used with the preceding approximation which would help to make calculations simpler.

2. TUTORIAL EXAMPLE UNRELIABILITY CALCULATIONS

Through use of the steps outlined in the previous section, the following bounds were obtained for the unreliability for the tutorial example (Section II-3):

$\overline{\rho}_{PLB}$	=	1.64928×10^{-2}
$\overline{\rho}_{PLB\text{-}CC}$	=	1.64909×10^{-2}
$\overline{\rho}_{PRF}$	=	1.64866×10^{-2}
$\overline{\rho}_{PRF\text{-}CC}$	=	1.64848×10^{-2}
Minimal Cut Upper Bound	=	1.59629×10^{-2}
$\overline{p}$ (exact)	=	1.59306×10^{-2}

IV. A BOILING WATER REACTOR (BWR) PHASED MISSION PROBLEM

The following sections contain an example boiling water reactor phased mission problem. Both exact and approximate solutions are given.

1. PROBLEM DESCRIPTION

As an example of the phased mission, the emergency core cooling system (ECCS) of a boiling water reactor (BWR) is considered. The ECCS used in this example problem is shown in Figure 14. It consists of eight subsystems which will be considered as components for this analysis. These components are:

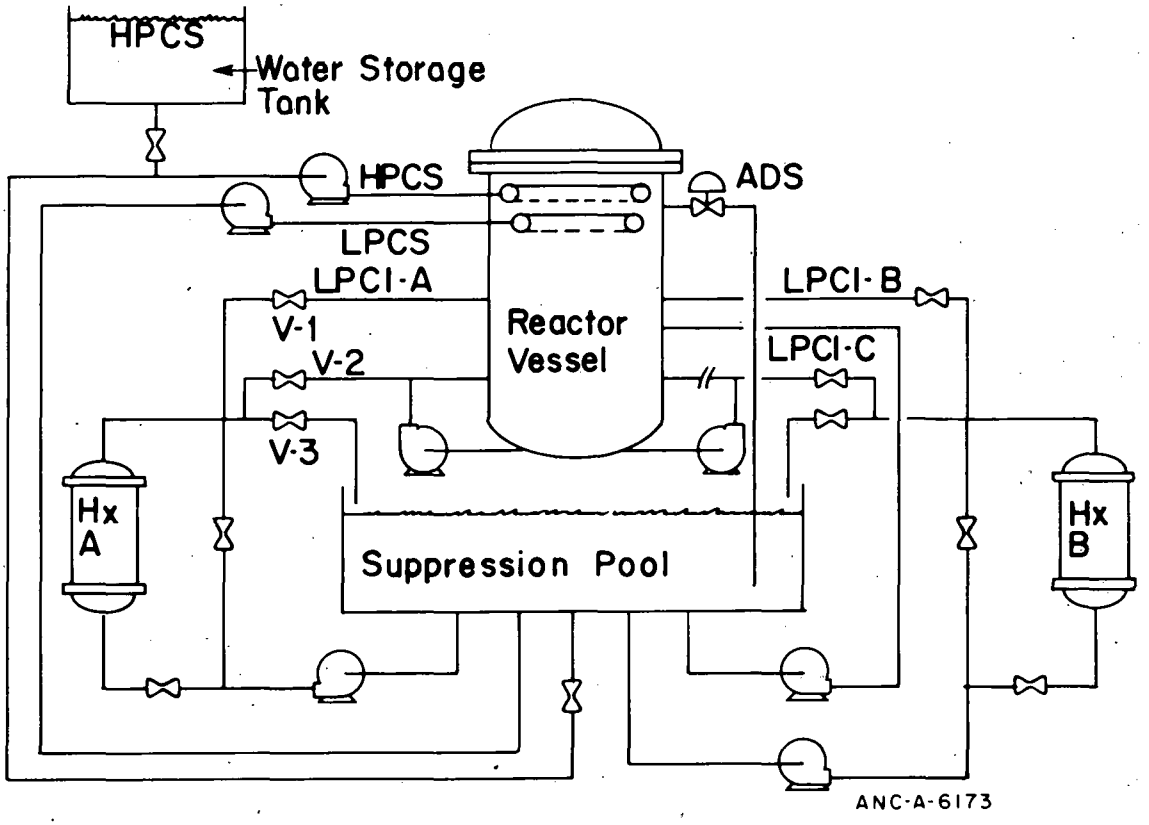


Fig. 14 BWR example emergency core cooling system.

High Pressure Core Spray (HPCS) System

Low Pressure Core Spray (LPCS) System

Three Low Pressure Core Injection (LPCI) Systems

Automatic Depressurization System (ADS)

Two Heat Exchangers (HX).

As seen from Figure 14 the two heat exchangers, denoted by HX-A and HX-B, are in the LPCI loops denoted by LPCI-A and LPCI-B.

One mission of the ECCS is to prevent excessive heating of the fuel rods within the reactor vessel as soon as possible after a large loss-of-coolant accident (LOCA) and then keep water circulating to and from the reactor vessel until the rods are cool.

After an LOCA has occurred, three phases for the ECCS can be identified. These are:

Phase 1 - Initial Core Cooling

Phase 2 - Suppression Pool Cooling

Phase 3 - Residual Heat Removal.

Each phase will be discussed briefly.

For the initial core cooling phase either the HPCS alone or the ADS and one of the low pressure systems are needed. The purpose of this phase is to reflood the core and cool the fuel rods as soon as possible after the break. Valve V-1 in Figure 14 is open during this phase and Valves V-2 and V-3 are closed. Here the phase was assumed to last one-half hour.

For the suppression pool cooling, Phase 2, the ADS is required to limit pressure buildup in the reactor vessel. One heat exchanger and the corresponding LPCI is needed to cool the water within the suppression pool. Finally, one of the three remaining low pressure systems or the HPCS is needed to circulate the water from the suppression pool to the reactor vessel. In this phase, Valve V-3 would be open and the other valves would be closed for suppression pool cooling. The length of this phase was taken as 36 hours.

For Phase 3, residual heat removal, the assumption is made that the break could be repaired or isolated so that the system could operate normally. Thus, one of the heat exchangers and the corresponding LPCI system are needed. Valve V-2 will be open, and Valves V-1 and V-3 will be closed (the complete flow loop is not shown). This phase is assumed to last 84 hours.

The fault tree for the TOP event "ECCS fails to cool core following LOCA" is shown in Figure 15. The component subscript denotes the transformation as discussed in Section II-2.

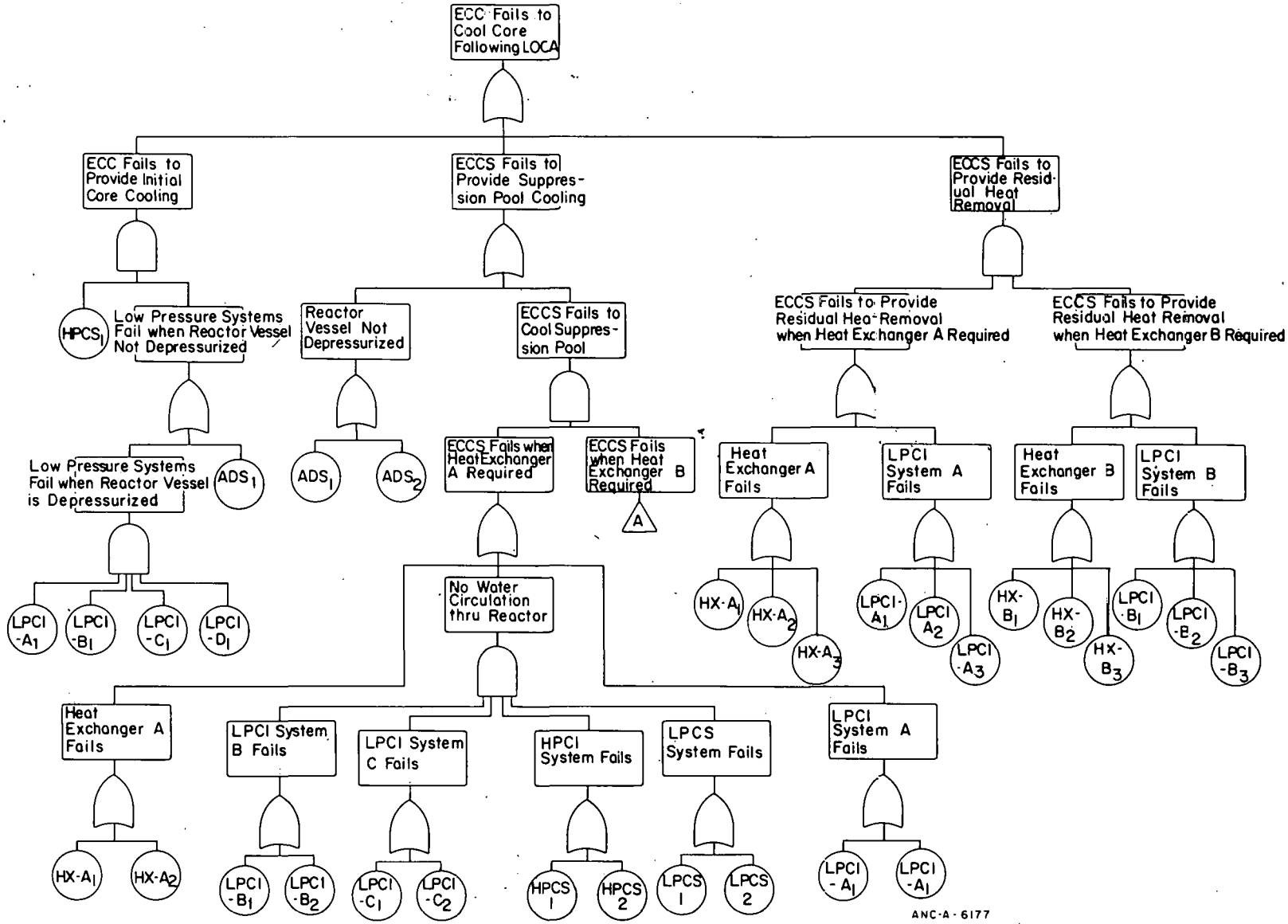


Fig. 15 BWR ECCS fault tree.

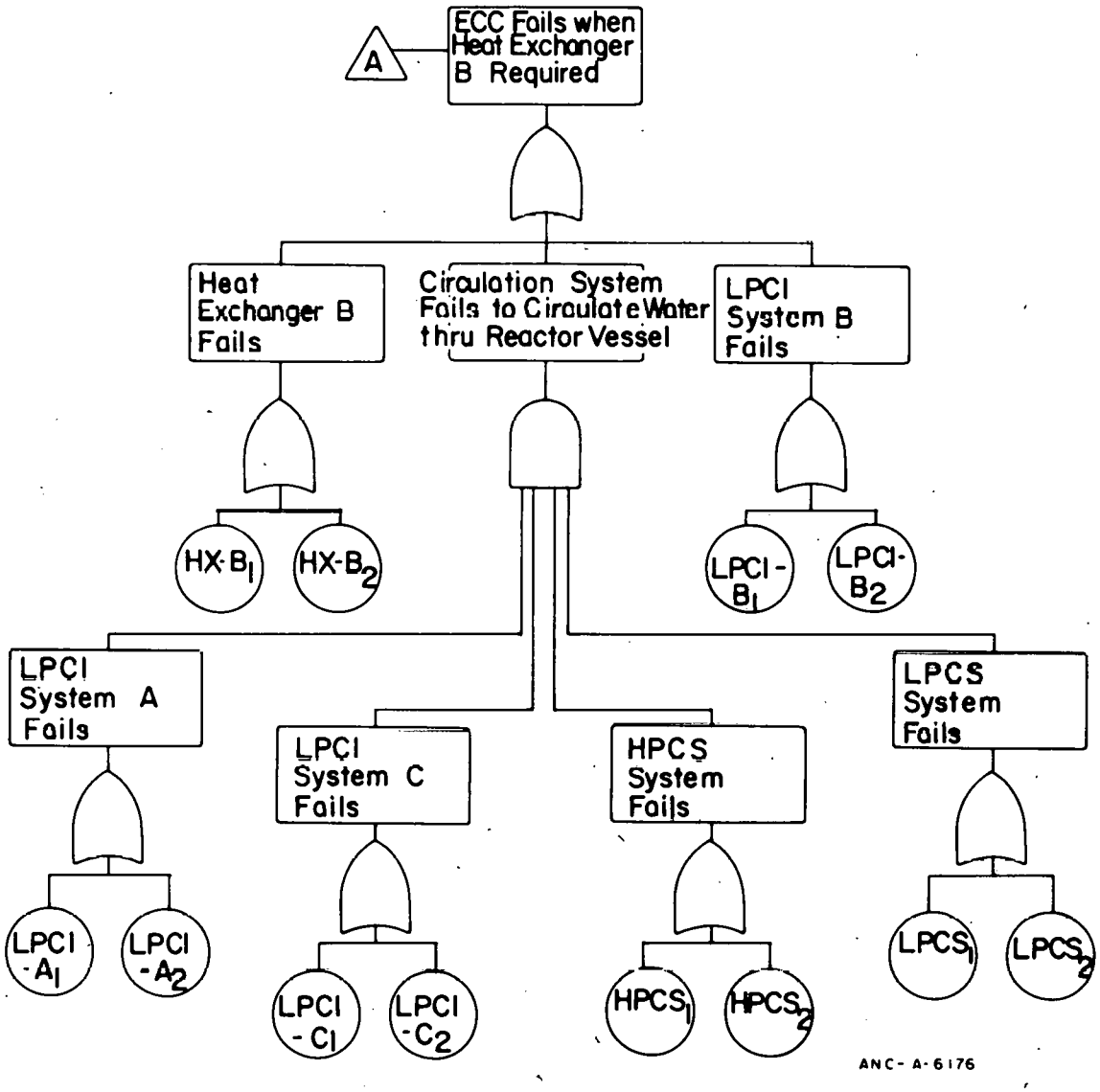


Fig. 15 BWR ECCS fault tree (contd.).

2. THE EXACT SOLUTION

From the fault tree in Figure 15, minimal cut sets for each phase and for the mission are obtained using the MOCUS computer program.

The minimal cut sets for each phase before cut set cancellation are:

Phase 1	{HPCS, ADS}, {HPCS, LPCI-A, LPCI-B, LPCI-C, LPCS}
Phase 2	{ADS}, {LPCI-A, LPCI-B}, {LPCI-A, HX-B}
	{HX-A, HX-B}, {HX-A, LPCI-B}
	{LPCI-A, LPCI-C, LPCS, HPCS}
	{LPCI-B, LPCI-C, LPCS, HPCS}
Phase 3	{HX-A, HX-B} {HX-A, LPCI-B}
	{LPCI-A, HX-B} {LPCI-A, LPCI-B}

After cut set cancellation only seven cut sets remain, which are:

Phase 1	None
Phase 2	{ADS}, {LPCI-A, LPCI-C, LPCS, HPCS}
	{LPCI-B, LPCI-C, LPCS, HPCS}
Phase 3	{HX-A, HX-B} {HX-A, LPCI-B}
	{LPCI-A, HX-B} {LPCI-A, LPCI-B}.

After the transformation is made, 70 minimal cut sets are obtained. No cancellation can be made after transformation.

The failure rates for each of the basic events used in this analysis are:

$\lambda_{ADS} = 1.4 \times 10^{-5}$

$\lambda_{LPCI} = 2.5 \times 10^{-5}$

$\lambda_{HPCS} = 2.7 \times 10^{-4}$

$\lambda_{LPCS} = 2.6 \times 10^{-6}$

$\lambda_{HX} = 2.8 \times 10^{-6}$.

These values should not be construed as realistic of the failure rates of the basic events used in this example. Instead, they are presented to illustrate the methodology.

The exact mission unreliability is calculated to be 5.22046×10^{-4} using the KITT-1 computer program.

3. APPROXIMATE SOLUTIONS

For the BWR example the following bounds on the mission unreliability were calculated:

$\overline{\rho}_{PLB}$	=	5.23084×10^{-4}
$\overline{\rho}_{PRF}$	=	5.23077×10^{-4}
Minimal Cut Upper Bound	=	5.22068×10^{-4}
$\overline{\rho}_{PLB\text{-}CC}$	=	5.22056×10^{-4}
$\overline{\rho}_{PRF\text{-}CC}$	=	5.22048×10^{-4}
$\overline{p}$ (exact)	=	5.22046×10^{-4}

A plot of the time-dependent mission unreliability is given in Figure 16. The graph illustrates a situation which can occur in a phased mission; that is, a discontinuity occurs in the unreliability at a phase boundary. Such discontinuities occur because of changes in system configuration.

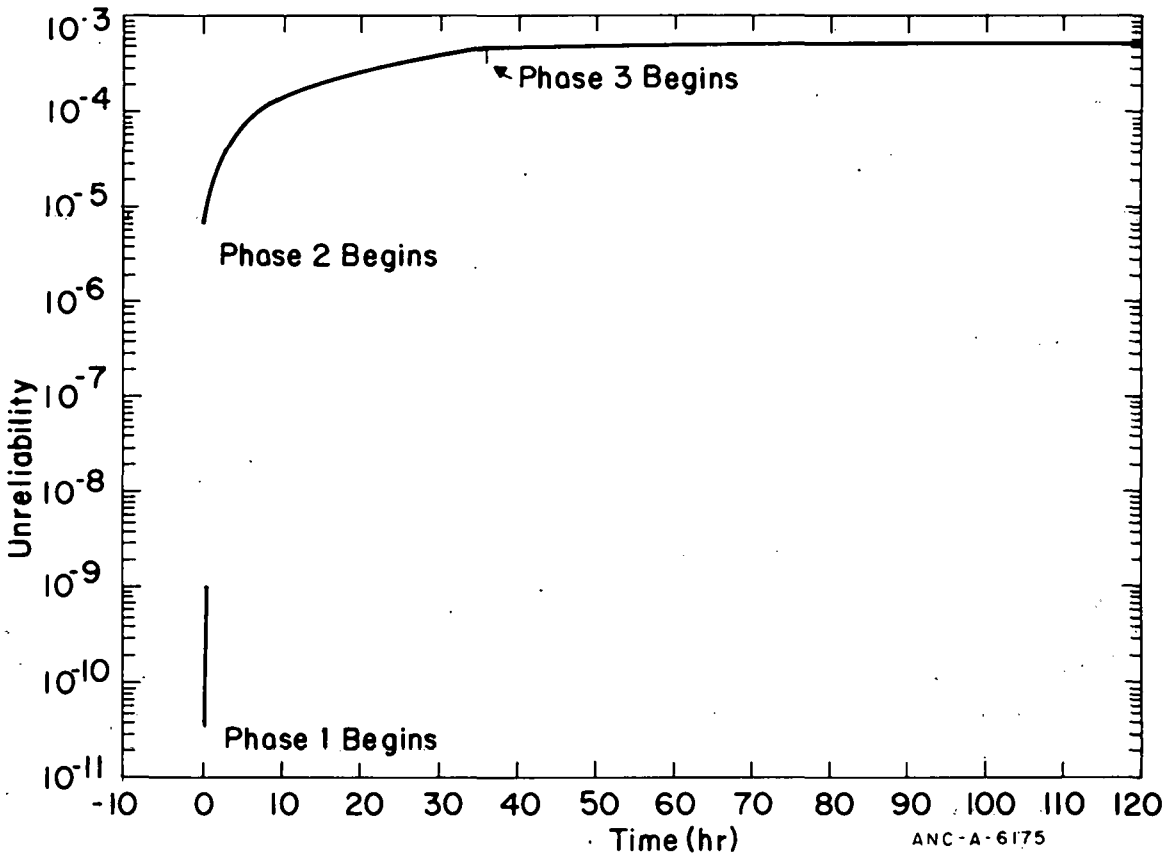


Fig. 16 BWR example unreliability graph.

V. CONCLUSIONS AND RECOMMENDATIONS

The phased mission method of Esary and Ziehms has provided an exact solution standard against which the rationale or theory behind any other proposed technique can be compared. Indeed, for small systems, the exact solution of Esary and Ziehms is obtainable for numerical comparisons of results. In cases for which large numbers of phases or components prevent obtaining an exact solution, the proposed approximation techniques presented in this report are theoretically sound and implementable in increasing order of difficulty $\overline{\rho}_{PLB\text{-}CC}$ $\overline{\rho}_{PLB}$ $\overline{\rho}_{PRF\text{-}CC}$ $\overline{\rho}_{PRF}$. The accuracy of approximation is in the order given in Section III-1.

The conclusion is that the approximation techniques for unreliability and unavailability estimates for phased missions, contained in this report, have a great deal of potential utility in the U.S. nuclear power industry. The techniques are not only implementable, but with slight modification, existing computer aids may be employed to the fullest. Without modification, the use of existing computer programs such as MOCUS, PREP, and KITT is possible (they were used in the preparation of this report) although such use would not be as efficient as could be.

The following recommendations are made:

(a) Prior to the use of these methods for tasks larger than those involving the examples of this report, effort must be expended to modify existing computer programs for their full utilization.

(b) An example of the application of these methods to a liquid metal fast breeder reactor (LMFBR) situation should be investigated. These methods should be utilized in a risk analysis of the Fast Flux Test Facility (FFTF) and Clinch River Breeder Reactor (CRBR).

(c) The phased mission unreliability and unavailability estimation methods, discussed herein, should be applied to a more complex nuclear power plant situation. The situation could be a loss-of-coolant accident risk assessment. The events causing the event tree dichotomies, that involve phased missions, could be identified; then, using methods of this report, probabilities of occurrence of these events could be estimated and compared with results obtained by other means. The risk numbers themselves should be obtained and compared.

(d) The task of (c) could be carried out also utilizing other techniques such as a combination of the phased mission methods together with cause-consequence analysis (Report V in this document).

(e) These methods should be disseminated to other reliability and safety analysts in the nuclear industry.

VI. REFERENCES

1. H. Ziehms, *Reliability Analysis of Phased Missions,* Thesis, Naval Postgraduate School, Monterey, Calif. (December 1974).

2. J. D. Esary and H. Ziehms, "Reliability Analysis of Phased Missions", *Conference on Reliability and Fault Tree Analysis, University of California at Berkeley, September 1974.*

3. E. J. Muth, "Reliability Assessment of Multiphase Missions", Apollo Support Department, G.E. Company, Daytona Beach, Florida.

4. J. H. Schmidt and S. A. Weisberg, "Computer Technique for Estimating System Reliability", *Proceedings of 1966 Annual Symposium in Reliability,* IEEE 7C26, pp 87-97.

5. *Reliability Evaluation Program Manual,* U.S. Navy Strategic Systems Projects Office NAVORD OD 29304 Revision A (1973) pp 2.8-3.8.

6. J. D. Esary and A. W. Marshall, "System Structure and the Existence of System Life", *Technometrics, 6* (1969) pp 459-462.

7. J. D. Esary and F. Proschan, "Coherent Structures of Nonidentical Components", *Technometrics, 5* (1963) pp 191-209.

8. J. B. Fussell, "Fault Tree Analysis -- Concepts and Techniques", NATO Advanced Study Institute on Generic Techniques of System Reliability Assessment, Liverpool, England (July 1973).

9. J. B. Fussell, "How to Hand-Calculate System Reliability and Safety Characteristics", *IEEE Transactions on Reliability, R-24,* 3 (August 1975).

10. D. F. Haasl, "Advanced Concepts in Fault Tree Analysis", *System Safety Symposium, June 8-9, 1965, Seattle, The Boeing Company.*

11. J. B. Fussell, E. B. Henry, N. H. Marshall, *MOCUS - A computer Program to Obtain Minimal Sets from Fault Trees,* ANCR-1156 (August 1974).

12. W. E. Vesely and R. E. Narum,*PREP and KITT: Computer Codes for the Automatic Evaluation of a Fault Tree,* IN-1349 (August 1970).

13. M. L. Shooman, *Probabilistic Reliability: An Engineering Approach,* New York: McGraw-Hill Book Company, Inc., 1968.

14. J. D. Esary and F. Proschan, "A Reliability Bound for Systems of Maintained, Interdependent Components", *Journal of the American Statistical Association, 65* (March 1970) pp 329-338.

REPORT V

ON THE ADAPTATION OF CAUSE-CONSEQUENCE ANALYSIS TO U.S. NUCLEAR POWER SYSTEMS RELIABILITY AND RISK ASSESSMENT

G. R. Burdick
J. B. Fussell

ABSTRACT

Cause-consequence analysis is a method of system reliability and risk analysis and is a combination of several methods presently used for analysis of U.S. nuclear power systems. Its advantages include providing the analyst a means for displaying the complex interrelationships among consequences and their causes. Cause-consequence analysis has been used as an aid in nuclear power plant reliability and risk assessment in Scandinavian countries since its inception in 1971. This report is both an investigation of cause-consequence analysis and a first step in adapting it to standardized use in the U.S. nuclear power industry.

ACKNOWLEDGMENTS

We gratefully acknowledge the many helpful suggestions we received from our colleagues during the preparation of this report. Special thanks go to R. J. Crump, R. H. Jennings, D. M. Rasmuson, M. E. Stewart, J. E. Trainer, J. R. Wilson, and J. C. Zipperer of the Systems Analysis Branch, Aerojet Nuclear Company.

CONTENTS FOR REPORT V

FIGURES

TABLES

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

ON THE ADAPTATION OF CAUSE-CONSEQUENCE ANALYSIS TO U.S. NUCLEAR POWER SYSTEMS RELIABILITY AND RISK ASSESSMENT

I. INTRODUCTION

The fields of reliability and safety analysis and design engineering need improved highly descriptive, diagrammatical methods for the combined tasks of discovery and investigation of possible undesirable situations in a complex, dynamic system. This need arises naturally from the intricacy of modern systems coupled with their increased potential for hazard if large sources of energy are involved such as in commercial nuclear power plants. Block diagrams, reliability diagrams, fault trees, and event trees are some of the techniques thus far generated to fill this need[a]. Each technique has advantages and disadvantages, the degrees of which are dependent upon the extent of complexity of the system under consideration and the amount and kind of information the analyst expects the method to produce. In addition, the methods vary greatly in the degree to which they aid the analyst in gaining a broad overview of a system as well as insights into its more subtle aspects.

The purpose of this report is to explain and illustrate the use of a relatively new diagrammatical method of describing failure sequences in complex systems, called cause-consequence analysis (CCA). CCA is related to failure modes effects and criticality analysis (FMECA), uses fault trees, and every cause-consequence diagram (CCD) has embedded in it a corresponding event tree. It is a descriptive tool and as such is a means for graphically linking together the various causes for undesirable events and the consequences of those events.

An attempt to construct an exact description of the events transpiring in a large system with components sharing common causes of failure[b] such as common physical properties, common defective manufacture, common operator error, common environment, common subcomponent failure , . . . , may generate a multitude of accident sequences and involve the analyst in a nearly endless and possibly unproductive exercise. One of the devices used in the Reactor Safety Study [2] to eliminate such potentially fruitless efforts was the event tree (18, Appendix I). Event trees are closely related to an analysis technique to be discussed herein.

CCA does not inherently ferret out common cause failures. It does, however, aid in common cause analysis and also has the ability to aid the analyst in obtaining a deeper understanding of the intricacies of the system. Its value lies in its fertility in producing combinations of failures, events, and consequences in a readily traceable, logical diagram which lends itself to meaningful quantitative as well as qualitative risk analyses. The CCD thus extends the event tree methodology.

---

[a] A good survey of methods is provided by H, E. Lambert[1].

[b] Report II in this document.

Section II contains a general description of CCA and the CCD. A tutorial example is employed to illustrate the technique. Section II-5 deals with the relationship between the CCD and event trees. Steps to follow in the construction of the CCD are provided in Section III. An example of the use of CCA in a liquid metal fast breeder reactor (LMFBR) consequence probability calculation and risk assessment is presented in Section IV.

Throughout, the material is presented on a level understandable by one having a basic knowledge of fault trees and probability theory. References providing additional background are given where appropriate. Sections that may be omitted by more advanced readers are pointed out as such.

II. CAUSE-CONSEQUENCE ANALYSIS

Cause-consequence analysis is a combination of fault tree analysis (cause) and inductive analysis (consequence). The inductive analysis is reported in a newly developed graphical form[a]. The following sections describe the form and the application of the technique of CCA through the use of illustrative examples.

1. GENERAL DESCRIPTION

The cause portions of the cause-consequence diagram are fault trees with the TOP events being component or system failures that can lead to various levels of undesired consequence depending on the degree of mitigation imposed by standby systems. The consequence portion of the diagram illustrates the array of consequence levels as a function of the binary state (failed or unfailed) of the standby system. The diagram, complete with cause and consequence portions, is referred to as the cause-consequence diagram. The well constructed CCD provides a clear but detailed flow chart which illustrates system interrelationships that either preclude or contribute to the probabilities of occurrence of the various consequences possible to arise from a particular main TOP event called an "initiating" event.

Symbols used in the cause portion are given in Figures 1 and 2. Symbols used in the consequence portion of the CCD are given in Figure 3. Use of the symbols of Figures 1, 2, and 3 are demonstrated in a sample CCD in Figure 4. References 9, 10, and 11 present a further discussion of fault trees and fault tree analysis.

2. A TUTORIAL EXAMPLE

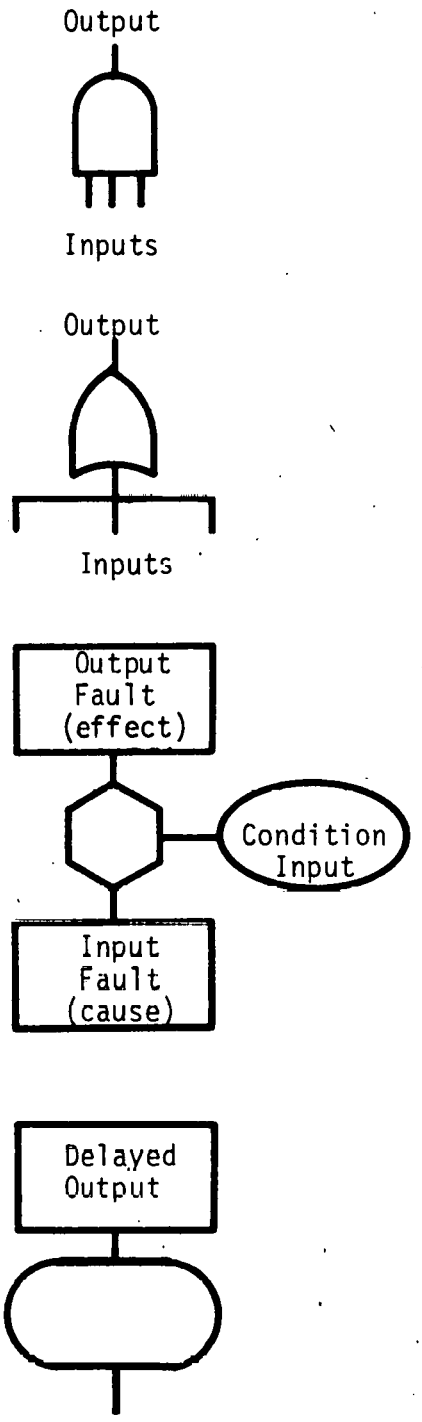
The following example is presented to demonstrate some of the fundamental aspects of CCA. The sample system is that of Figure 5. The motor is located such that it has a chance of causing a catastrophic fire. Figure 6 is the CCD for this example situation.

The initiating or trigger event is "motor overheats". This is the TOP event to the fault tree at the bottom of Figure 6[b]. This fault tree is the primary cause portion of the CCD.

---

[a] The CCA methodology was developed by D. S. Nielsen[3,4,5] of the Danish Atomic Energy Commission. The format and symbology used here has been modified to be consistent with that used in the United States nuclear power industry. Taylor[6,7,8] presents additional information.

[b] A complete discussion of the fault tree development for this example was given by Fussell[9].



AND Gates

Coexistence of all inputs required to produce output.

OR Gates

Output will exist if at least one input is present.

INHIBIT Gates

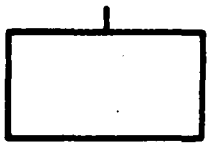
Input produces output directly when conditional input is satisfied.

DELAY Gates

Output occurs after specified delay time has elapsed.

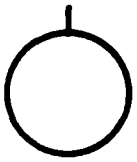
Fig. 1 Fault tree logic symbol.

The consequence portion of the CCD unfolds, to reflect the sequence of events that could be encountered by the system, beginning with the initiating event, and develops using branching operators, time delays, OR gates, inverse AND gates, with event descriptor tags



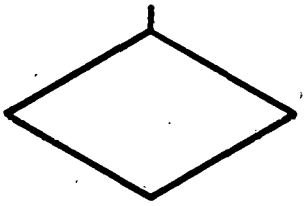
RECTANGLE

A fault event resulting from the combination of more basic faults acting through logic gates.



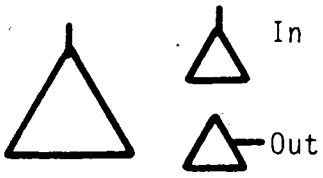
CIRCLE

A basic component fault - an independent event.



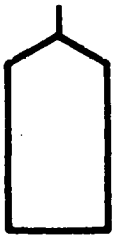
DIAMOND

A fault event not developed to its cause.



TRIANGLE

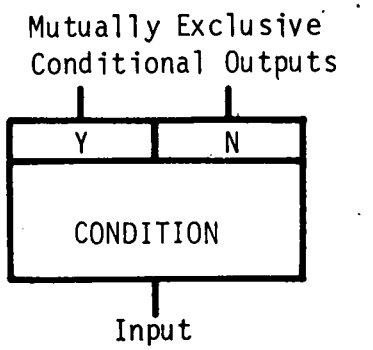
A connecting or transfer symbol.



HOUSE

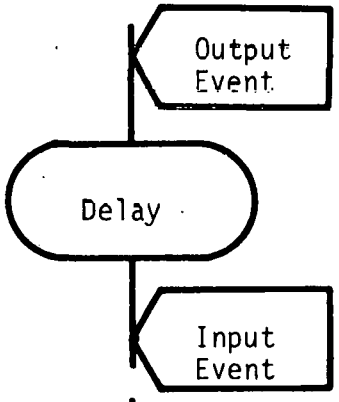
An event that is normally expected to occur or to never occur. Also useful as a "trigger event" for logic structure change within the fault tree.

Fig. 2 Fault tree event symbols.



BRANCHING OPERATOR

Output is "yes" if condition is met; "no" otherwise.

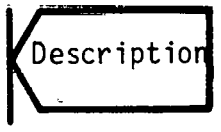


DELAY OPERATOR

Indicates the amount of time delay required for output event to result from the input event.

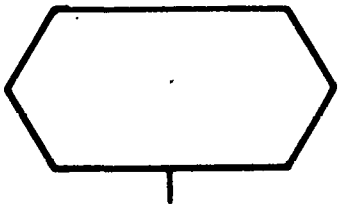
DIRECTOR

Indicates the direction of event flow.



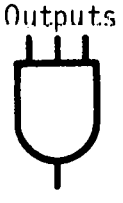
EVENT DESCRIPTOR

Describes the event present at specified position in chart.



CONSEQUENCE DESCRIPTOR

Describes the consequence. A terminal symbol.



Inverse AND Gate

All outputs occur if the input occurs.

Fig. 3 Consequence diagram symbols.

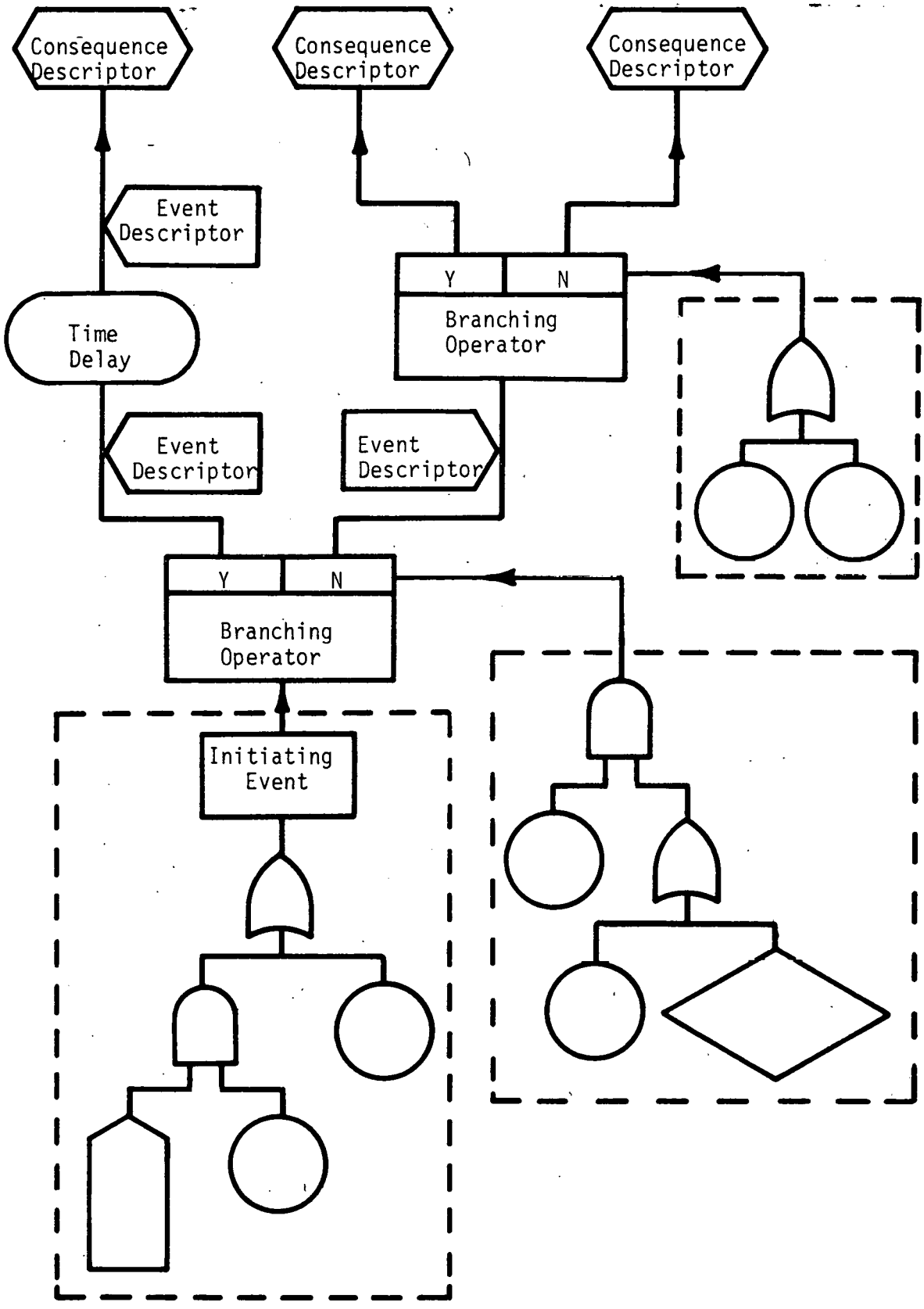
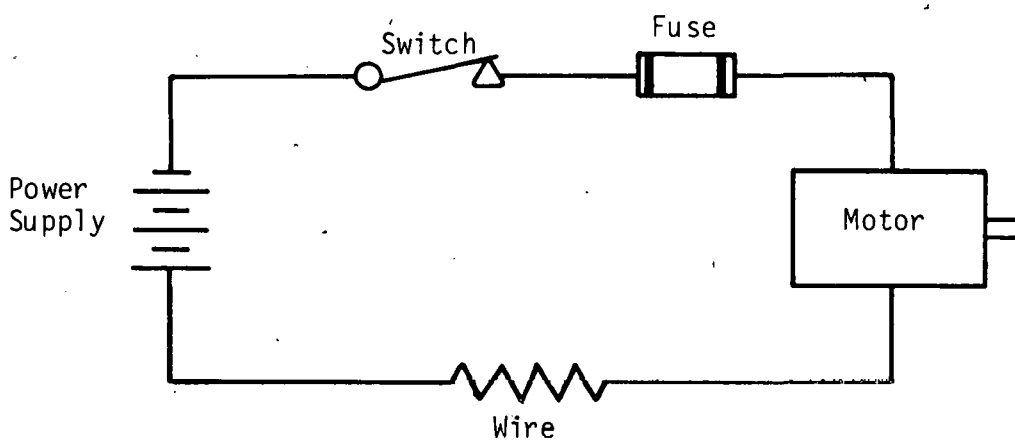


Fig. 4 Sample cause-consequence diagram.



TOP event	=	Motor overheats
Initial condition	=	Switch closed
Not-allowed events	=	Failures due to effects external to system
Existing events	=	Switch closed

Fig. 5 Sample system.

appropriately placed to add clarity. This process continues until each path ends in a consequence description. Each branching operator has its attendant fault tree which develops causes for the undesirable condition in that operator. Common cause failure exposure potential originates among these fault trees for branching operators and the initiating event.

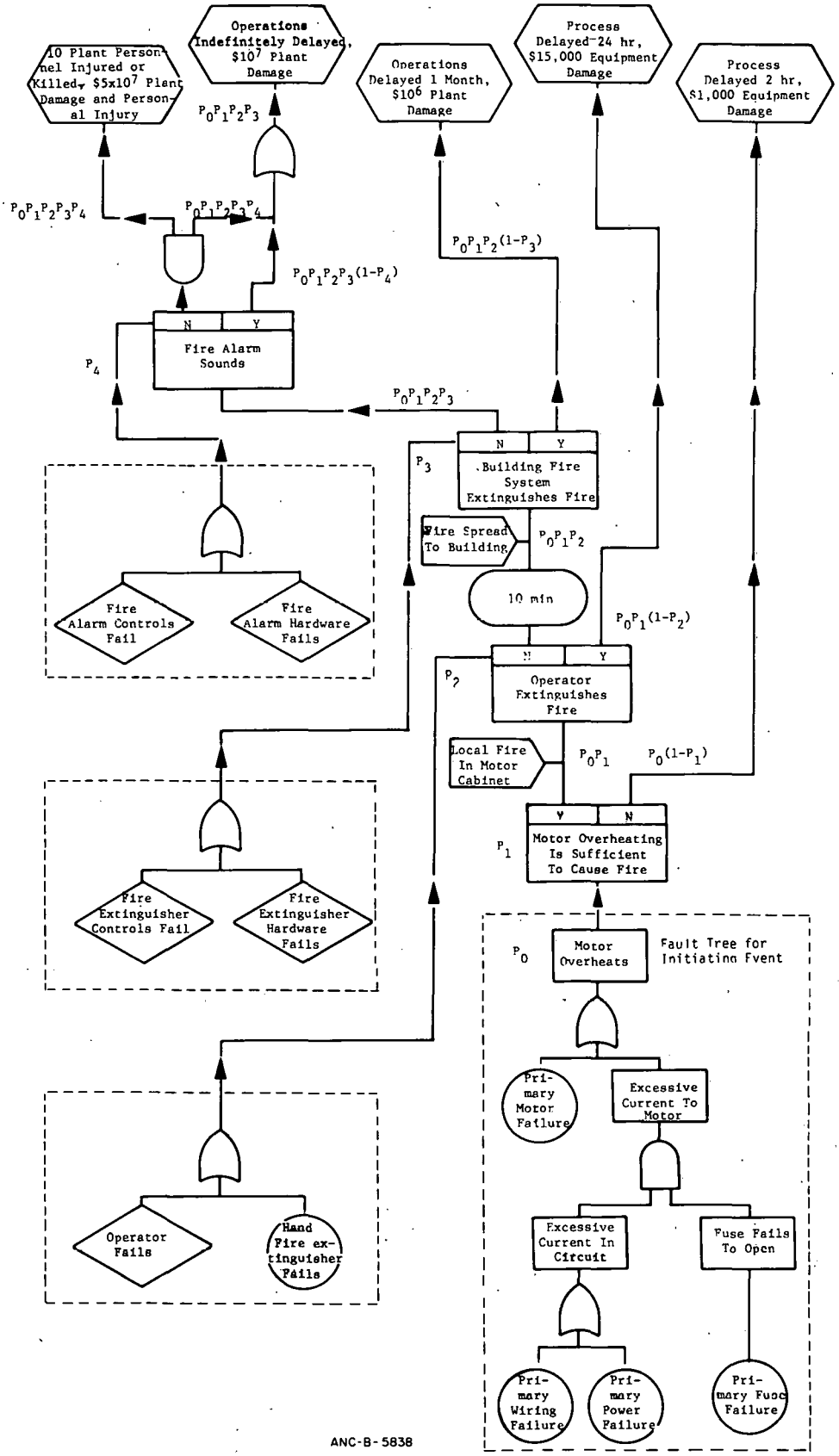


Fig. 6 Sample system cause-consequence diagram.

3. CALCULATION OF CONSEQUENCE PROBABILITIES AND RISK ASSESSMENT OF CONSEQUENCES

Knowing the probability of occurrence of the basic events in each fault tree, one may then calculate, or obtain an upper bound for, the probability of occurrence of each fault tree TOP event in the CCD by using methods developed by Vesely[11] et al. If the branching operators are statistically independent, then by multiplying probabilities at each branching operator, corresponding to event occurrence (yes) or nonoccurrence (no), one eventually obtains an estimate of the probability of occurrence of each of the consequences in the CCD. How good this estimate is, of course, depends upon the number and kind of common cause failures involved, the degree of accuracy of the probabilities of occurrence of all basic events, and the minimization of oversights and omissions in the associated fault trees.

Those unfamiliar with fault trees, event trees, and the calculation of probabilities for sequences of events or for TOP events should trace through the following discussion of the calculation of the probabilities of occurrence of each consequence for the tutorial example. More experienced analysts may proceed to Section II-4.

The preceding process will now be followed through for the CCD of Figure 6 using the probabilities, listed in Table I. Probabilities[a] p_0, p_2, p_3, p_4 are assumed to be constant and have been previously calculated or estimated from the fault tree having as its TOP event the corresponding event or condition listed in the table. Probability p_1 in this example is assumed to be a number provided not by a fault tree but estimated from experience or perhaps obtained from actual failure records. Probabilities are valid for exactly a one-year operating cycle. Branching operator events are assumed to be independent.

TABLE I

PROBABILITIES OF OCCURRENCE FOR THE EVENTS AND CONDITIONS IN THE CDD OF FIGURE 6

Event	Probability
Motor overheats	$p_0 = 10^{-3}$
Motor overheating sufficient to cause fire	$p_1 = 10^{-1}$
Extinguisher fails to extinguish fire	$p_2 = 10^{-2}$
Building fire system fails to extinguish fire	$p_3 = 10^{-2}$
Fire alarm fails to sound	$p_4 = 10^{-3}$

[a] Reference 12 discusses introductory probability theory.

The calculation of the probability of occurrence of each of the consequences begins with the probability of occurrence of the initiating event, p_0. The flow of events from the initiating event, "motor overheats", leads to and only to the branching operator containing the condition "motor overheating is sufficient to cause fire". The "yes" probability of occurrence of this condition, p_1 , multiplied by p_0 p_1 of having a local fire in the motor cabinet. The "no" probability of occurrence of this condition $(1-p_1)$ multiplied by p_0 gives the probability $p_0(1-p_1)$ of reaching the consequence "process delayed 2-hours -- $1,000 equipment damage". Along the path past the descriptor "local fire in motor cabinet", another branching operator is reached for which the probability p_2 is factored in for the undesirable condition. For the desirable condition $(1-p_2)$ is factored in the same manner as with the previous branching operator. p_0 p_1 $(1-p_2)$ is then the probability that the events "motor overheats", "motor overheating is sufficient to cause fire", and "operator extinguishes fire" all occur. p_0 p_1 $(1-p_2)$ is obviously the probability that the first two events occur but the third does not. This latter case leads to the consequence "process delayed 24 hours -- $15,000 equipment damage".

Continuing up the left-hand flow path, the constructor of the chart has determined that, after a ten minute delay, a fire will be spreading to the building. (A situation such as this is an obvious clue to a design engineer that a need for additional protective equipment exists.) Calculation of probabilities through the next two branching operators follows the same procedure as before. Nothing unfamiliar is encountered until the path leaves the branching operator "fire alarm sounds". The left-hand event flow path leads, with probability $p_0p_1p_2p_3p_4$, to an inverse AND gate whereas the right-hand event flow path leads, with probability $p_0p_1p_2p_3(1-p_4)$, to an OR gate. The other input to this OR gate comes from the inverse AND gate which gives a probability of occurrence $p_0p_1p_2p_3p_4$. Since outputs of branching operators are mutually exclusive events, the probability of occurrence of one of the inputs to the OR gate is simply the sum of the individual probabilities; that is, $p_0p_1p_2p_3$ is the probability of occurrence of the consequence "operations indefinitely delayed, 10^7 plant damage". The remaining two consequences have probability of occurrence $p_0p_1p_2p_3p_4$ since their event flow paths lead directly from the inverse AND gate. Table II is a listing of the consequences, their probabilities of occurrence, and the risk number for each.

CCA affords a tool with which to calculate risks associated with various consequences possible from a set of causes. The Reactor Safety Study [2] defines risk as:

$$\text{Risk}\left(\frac{\text{Consequence}}{\text{Unit time}}\right) = \text{Frequency}\left(\frac{\text{Events}}{\text{Unit time}}\right) \times \text{Magnitude}\left(\frac{\text{Consequences}}{\text{Event}}\right).$$

This expression is agreeable with the technique of CCA.

The example of Figure 5 and the CCD of Figure 6 are again considered. The product of consequence probability and damage level produces risk numbers for each consequence. Table II lists the consequences and their associated probabilities of occurrence and risk numbers when each hour of process delay is valued at $1,000. In the last case a risk number has no meaning.

TABLE II

CONSEQUENCE, PROBABILITY, AND RISK FOR THE

SAMPLE SYSTEM OF FIGURE 5

Consequence	Events/Yr	Risk($/Yr)
Process delayed 2 hr 10^3 equipment damage	10^{-3}	3.00
Process delayed 24 hr $15 x 10^3 equipment damage	10^{-4}	3.90
Operations delayed 1 mo 10^6 plant damage	10^{-6}	1.72
Operations delayed 1 yr 10^7 plant damage	10^{-8}	0.19
$5 x 10^6 damages for personnel injuries	10^{-11}	0.005
10 plant personnel injured or killed	10^{-11}	---

4. EVENT TREES AS IMPLIED BY CAUSE-CONSEQUENCE CHARTS

An event tree is a CCD with all fault trees, gates, descriptors, and delay operators removed. The branching operators are replaced by the branch points in the event tree; the mutually exclusive outputs of the branching operators provide the dichotomies which form the event tree branches. For example, the event tree of Figure 7 is obtained from CCD of Figure 6.

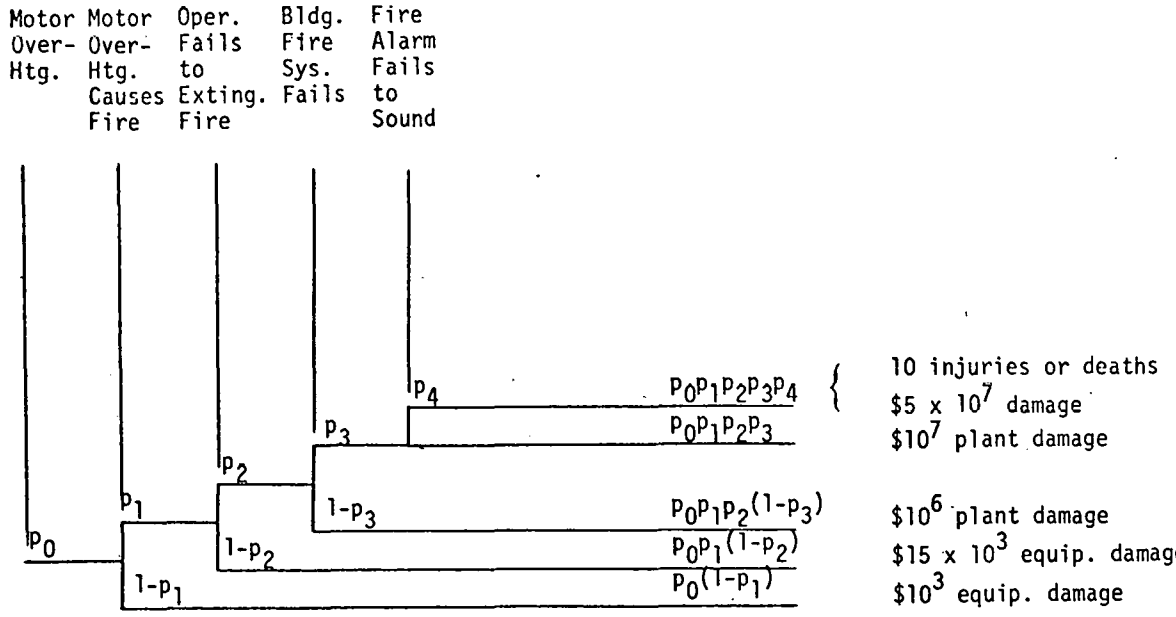


Fig. 7 Event tree for the CCD of Figure 6.

A more topical example is provided by an LOCA in a typical nuclear power plant. The CCD for this hypothetical situation is Figure 8 where the initiating event is a pipe break in the main coolant system of a pressurized water reactor (PWR). The CCD is simplified for illustrative purposes. Corresponding to this CCD is the event tree of Figure 9 which appeared in the article "The AEC Study on the Estimation of Risks to the Public from Potential Accidents in Nuclear Power Plants", by N. C. Rasmussen[13]. The event tree is obviously more streamlined than the CCD and serves the purpose of brevity well. However, the event tree contains far less information than a CCD. Consequently, to a systems analyst or design engineer, the event tree is not as rich in description as is the CCD.

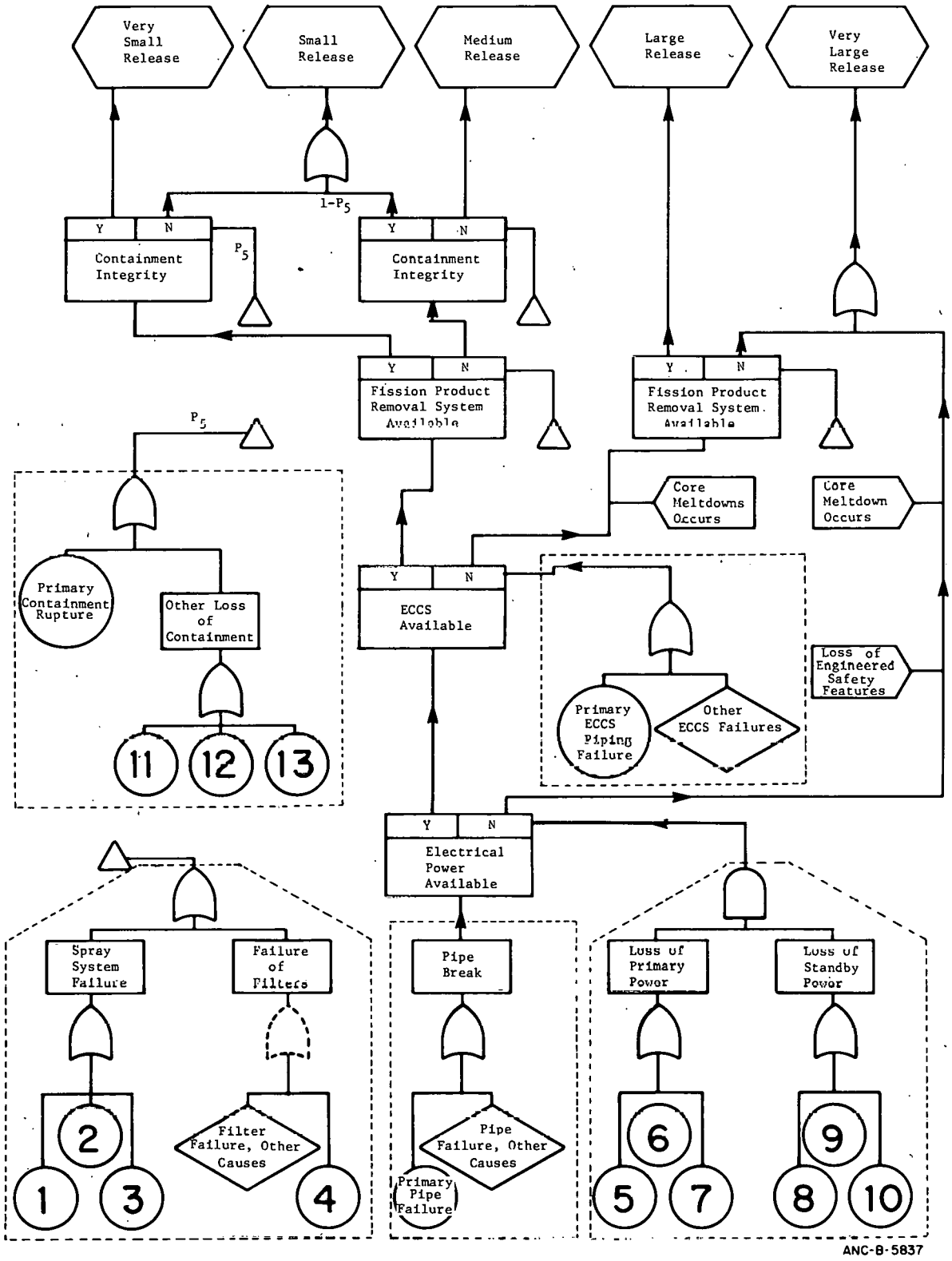


Fig. 8 CCD for hypothetical LOCA.

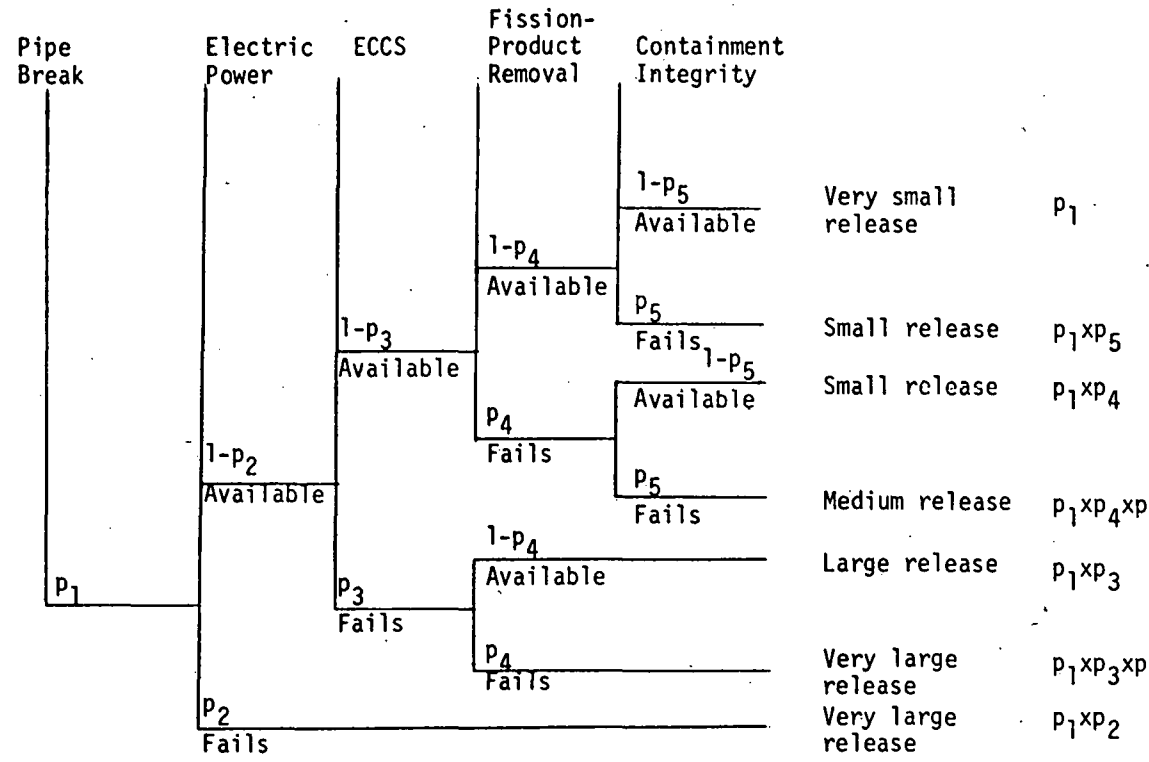


Fig. 9 Simplified event tree for an LOCA in a typical PWR.

III. A GUIDE TO CAUSE-CONSEQUENCE DIAGRAM CONSTRUCTION[a]

The previous examples were purposely simplified to illustrate fundamental properties of CCA, CCD, and event trees. For the complex systems encountered in practice, the entire process of CCA presupposes a great deal of knowledge not only of the components comprising the system, and their various failure modes, but also a detailed knowledge of the interrelationships among the components. Preliminary requirements and suggested steps in CCD construction are presented in the following sections.

1. PRELIMINARY REQUIREMENTS

The fundamental tool for construction of a CCD for a complex system is a functional block diagram of the system complete with indicators for directions of flow of information, fluids, current, or other quantities vital to proper system functioning. Supplementing the block diagrams should be blueprints of the physical location of the components and detailed wiring and piping diagrams and schematic drawings. Firsthand knowledge of the behavior of various components, at least under normal circumstances, is also of great use as is the knowledge of probable behavior under abnormal conditions. Specifically, the cause-consequence analyst should be familiar with the:

(1) Task the system is to perform

(2) Types of components in the system

(3) Component operating modes

(4) Dynamic relationships among components

(5) Physical location of components and subsystems

(6) Design limitations and failure modes of components

(7) Physical properties of materials used in or carried by the system

(8) Trip limits or set points of installed safety devices.

2. SUGGESTED STEPS IN CONSTRUCTION

Having satisfied the preceding requirements, the cause-consequence analyst is equipped to begin construction of the CCD for the system. Although no listing of steps

[a] The material in this section incorporates observations and remarks made by D. S. Nielsen[5].

could be totally pertinent to each analysis or completely inclusive for all analyses, the following steps are provided for inspection by potential analysts.

(1) One critical event consistent with a component or subsystem operating mode is selected.

(2) The dynamic model or block diagram is modified taking the critical event into account.

(3) The changes or transients (delay and magnitude) of the main system parameters at locations where protective devices or parts of protective devices (safety valves, sensors, . . .) are available are specified.

(3a) Which trip limits or set points are exceeded is determined.

(4) Whether loading limits for relevant system components are exceeded by effects from system parameter changes or transients is determined.

(5) The environmental changes within relevant areas, such as pressure, temperature, or radiation changes; missile potentials, flooding, and escape of materials are identified. (A consequence of environmental changes may be a critical event in other structurally and operationally separate systems.)

(5a) Potential transgressions of trip limits or set points (due to environmental changes) at locations outside the main system where protective devices or parts of protective devices (safety valves, sensors, . . .) are available are identified.

(5b) Whether conditions (temperature T_1, pressure P_1, concentration C_1, and presence of ignition source) are present for fire or explosion in case of escaped material is determined. If so, the potential, significant consequences ('damage to -- ', 'injury to staff') is determined.

(5c) Accident-limiting barriers, if any, designed to cope with environmental changes are identified.

(5d) Whether the environmental pressure or temperature changes and transients exceed the specified loading limits for the individual accident-limiting barriers, if any, are determined. If so, the potential, significant consequences are determined.

(6) Which 'designed protective actions' (that is, accident-preventing or -limiting actions) are potential according to the results of Steps (3a) and (5a) are identified. In this connection:

(a) A designed protective action can, if released, be 'desirable' as well as 'undesirable' in the context of the actual accident situation

(b) A desirable designed protective action may fail (that is, designed protective action "x" does not occur as intended).

(7) A consequence diagram is constructed which shows the potential combinations of 'released' and 'not released' designed protective actions.

(8) For each combination identified in Step (7), the dynamic model [Step (4)] is modified.

(9) For each of the identified potential accidents, the changes or transients of main system parameters (pressure, temperature) in relevant process components are specified.

(10) The following are determined for each of the identified, potential accidents: Whether loading limits for relevant process components are exceeded by effects from system parameter changes or transients. If so, the potential, significant consequences ('damage to – ', 'escape of — ', 'injury to —- ')are determined.

(11) The consequence search, if relevant, is continued; otherwise Step(2) is followed.

(12) Whether significant consequences are identified is determined. If so, Step (13) follows; otherwise Step (1) is repeated.

(13) The potential causes of the critical event are identified.

(13a) If the critical event is a failure mode of a 'static' component (pipe-line, flange, vessel, . . .), then the potential influences of other structurally and operationally separate systems (for example, effects of missiles, flooding, pressure, temperature, vibration, . . .) are identified.

(13b) If the critical event is a failure mode of an 'active' component (for example, 'control valve "x" closes', or 'pump "x" fails'), then the relevant, functionally related units and their locations are identified. For each unit, the relevant failure mode and the possible environmental effects that may cause it are identified.

(13c) The result is displayed in a cause diagram (fault tree) with reference to relevant information.

(14) Whether the individual system that is called upon to perform a desirable accident-preventing or accident-limiting action is capable of coping with the critical event is determined, assuming that no faults in the system have occurred or occur during accident conditions. For instance, the adequacy of the response time of the system is determined.

(14a) The potential 'in-system' causes of the failure 'designed protective action "x" does not occur as intended' (for example, an 'unannounced' basic fault event has occurred) are identified.

IV. AN LMFBR EXAMPLE

The example contained in this section is based on a hypothetical pot-type liquid metal fast breeder reactor (LMFBR) such as that described in the 1,000 MWe LMFBR Follow-On Study[14]. The purpose of the example is to show a possible type of application of CCA to a nuclear system. The resolution of the analysis is restrained in keeping with the illustrative nature of this report. Also, the event descriptions appearing in the CCD are highly abbreviated in keeping with the tutorial emphasis of this report with regard to methodology rather than application. The consequences shown in Figure 10 are absolutely fictitious. Frequent reference to the CCD of Figure 10, and the steps of Section III-2 are recommended concurrent with study of this example.

1. SYSTEM DESCRIPTION

A listing of subsystems and some of their typical components is provided in Table III. A basic block diagram of the system is given in Figure 11.

TABLE III

LMFBR EXAMPLE SUBSYSTEM AND COMPONENT TABLE

Subsystem	Components
Scram system	Operator, rod release mechanisms
Residual heat removal system	EM pumps, heat exchangers, tubing, motors, fans
Reactor building cooling system	Compressor, heat exchangers, tubing, fans, motors
Pot cover structure	Cover structure, plug, cover structure holddown components
Electrical systems	Main system, standby system
Reactor building isolation	Operator, monitors, miscellaneous isolation equipment
Reactor system	Pumps, motors, piping, reactor vessel, control rods, fuel rods, ...
Reactor control system	Operator, control rod drive mechanisms, miscellaneous electrical and electronic control equipment
Reactor monitoring system	Flow, temperature, neutron monitors

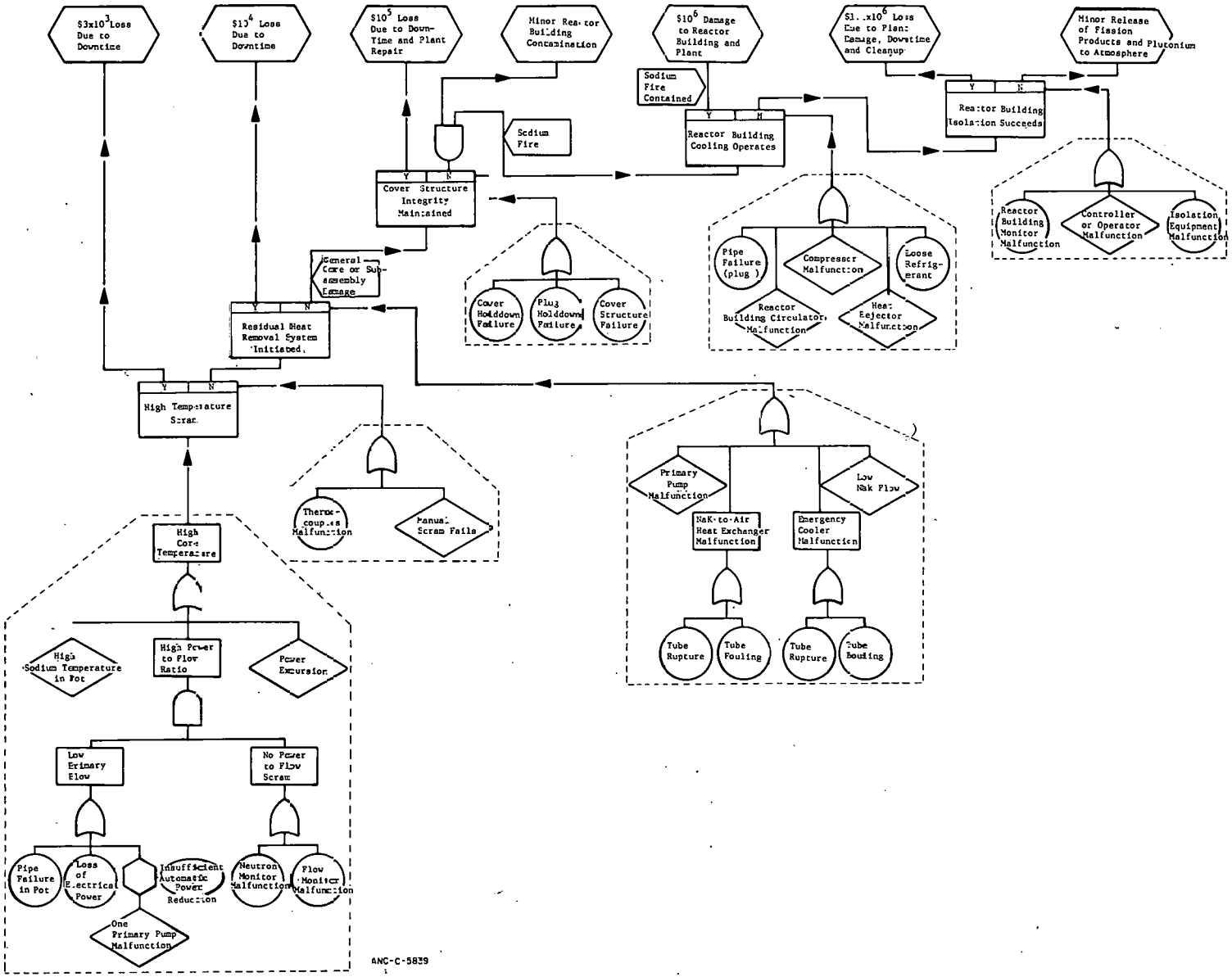


Fig. 10 CCD for LMFBR example.

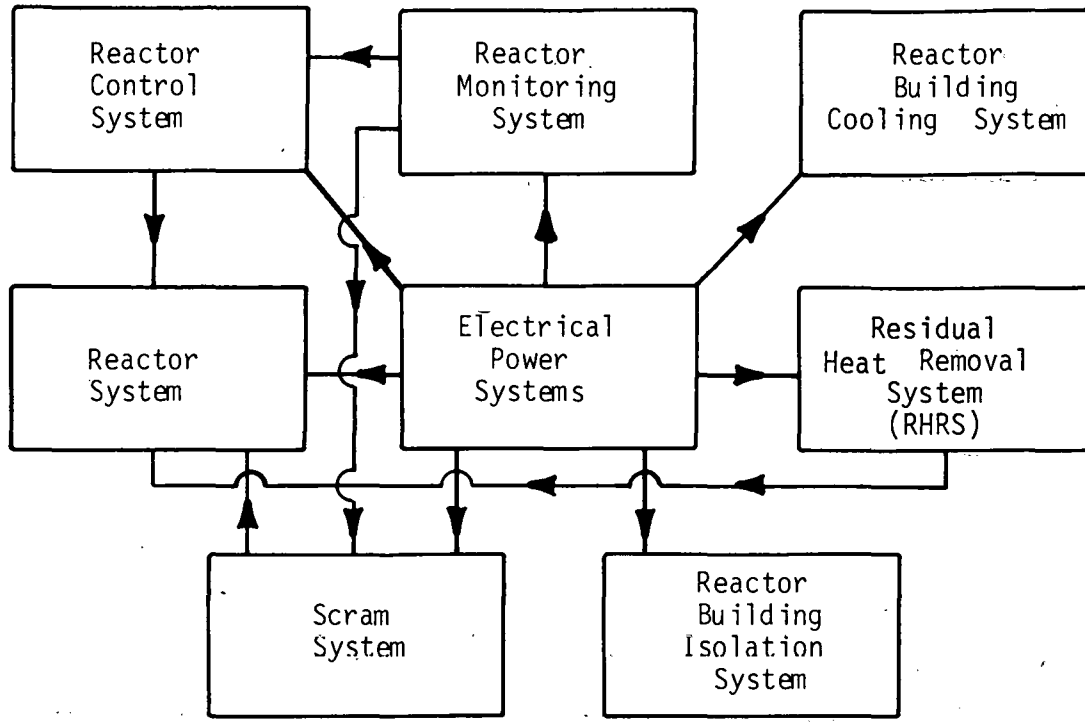


Fig. 11 LMFBR example system block diagram.

2. CONSTRUCTION OF THE CCD

The analyst chooses as his initiating event [Step (1)] "high core temperature" and his intent is to discover both the cause and the possible consequences of the occurrence of this event. (The assumption is that he satisfies the requirements of Section III-1 and has at his disposal all the requisite system information such as blueprints and wiring and block diagrams). The core temperature is further assumed to exceed the high temperature scram limit, and the analyst notes [Step (3a)] that thermocouples in the reactor monitoring system should sense this condition. The next step having any immediate bearing is Step (6), where the analyst notes that the thermocouples should initiate a scram directly and they should provide a high temperature alarm for an operator initiated scram. Step (7) provides the branching operator "High Temp Scram" and the consequence descriptor "$3 x 10^5 Loss Due to Downtime". Continuing to follow Step (7), the analyst notes that RHRS should have been initiated. Steps (7), (8), (9), and (10) produce the consequence descriptor "10^6 Loss Due to Downtime" and the branching operator "Cover Structure Integrity". The process is continued until the analyst exhausts the supply of both protective action events (remaining branching operators) and interesting possible consequences (remaining consequence descriptors). Steps (12) through (14c) then provide the cause portions (fault trees) for the CCD. The analyst is now ready to calculate probabilities of the various consequences and to make an assessment of their risk, as done for the tutorial example in Sections III-3 and III-4.

V. CONCLUSIONS

CCA is of major value in that it allows the analyst to work an otherwise unmanageable problem in segments. A standard approach to a typical problem has been to determine inductively the possible consequences and use these as TOP events for an array of fault tree analyses. The results of this approach are numerous and, in practice, intractable logic models.

One important advantage that CCA has over other techniques is that it provides, through the CCD, a better method for depicting the many logical combinations of events that contribute to a particular consequence or group of consequences. It helps the engineer or analyst to understand better his system by providing the means by which he can organize his knowledge. It further provides a model from which probabilities of occurrence of various consequences can be estimated and from which risk numbers may be obtained for the consequences without loss of causal information as with event trees.

In constructing the CCD, the analyst is given the option of working forward from an event or backward from a consequence, and he would probably do both. This capability is also a feature of failure modes and effects analysis (FMEA)[1]; however, the tabular format of FMEA is not as amenable to the rapid tracing of possible events and consequences as is the CCD diagrammatical format.

As a result of the feature of CCA that the problem under analysis is broken into segments, a most severe limitation of CCA is introduced. This limitation also applies to the event tree approach. The events described in the branching operators at a lower tier of the consequence portion of the diagram are assumed to be independent of all branching operator events at higher tiers. Since the converse is not a required assumption, ordering of branching operators is extremely important. Ordering these events from lower to upper tiers with respect to the expected sequence of occurrence of the events indicated by the branching operators is a good "rule of thumb" to mitigate the effects of the preceding assumption. Mitigation of these effects is also possible through skillful construction of the fault trees attached to the branching operators; however, details of all these techniques are beyond the scope of this report.

The newness of CCA could itself be a limitation. CCA combines several different analysis techniques and it therefore requires a broad scope of training. In combining the techniques of deductive analysis, decision and event tree analysis, fault tree analysis, and FMECA, as does CCA, the result is quite naturally a technique which is more complex than any of its parts. This complexity alone could be a problem, especially for an inexperienced analyst confronted with a very intricate system. Many manweeks may be required to construct a CCD for a complex system, even by an experienced analyst.

Common cause failures are a problem in CCA just as they are in any other known diagrammatical method. In the calculation of meaningful system characteristics, probabilities of occurrence of event, and consequences that are dependent upon common cause

failures, the systems analyst must still use appropriate approximation techniques such as those used in the Reactor Safety Study, Appendices I and IV [2].

Cause-consequence analysis should be applied to complex U.S. nuclear reactor systems in both safety and risk assessments, and the results should be compared with those obtained by other means. The other means could be FMECA, event trees coupled with fault trees, or fault trees alone. The comparison should be made to include not only judgments of the accuracy of numerical results but also of the desirability of the display and the rapidity of retrieval of information about the system. The CCA should be combined with other new methods of analysis such as the phased mission techniques[15].

The technique of CCA is not untried. It has been used since 1971 in Scandinavia[3,4,5,6,7,8,16]. So far, the only articles directly concerning or applicable to nuclear power plant reliability calculations using CCA to appear in U.S. technical journals are those submitted by the Danes[17,18]. However, the degree of respect that CCA already enjoys from recognized U.S. experts in the reliability field is quite high. This recognition is best illustrated by the enthusiastic editorial "Fault-Trees and Cause-Consequence Charts" by R. A. Evans, Editor, IEEE Transactions on Reliability[19]. In essential agreement with Evans, CCA being looked upon as a replacement for other techniques of reliability analysis is not recommended. CCA being added to the supply of methods, used where best suited, and perhaps in conjunction with other techniques is recommended.

VI. REFERENCES

1. H. E. Lambert, *Systems Safety Analysis an Fault Tree Analysis,* UCID-16238, Laurence Livermore Laboratory, Livermore, Calif. (May 9, 1973).

2. *Reactor Safety Study,* WASH-1400, U.S. Atomic Energy Commission, Washington, DC (August 1974).

3. D. S. Nielsen, *The Cause-Consequence Diagram Method as a Basis for Quantitative Accident Analysis,* RISØ-M-1374, Danish Atomic Energy Commission (1971).

4. D. S. Nielsen, *Cause-Consequence Diagrams,* NARS Publication 2, Nordic Working Group on Reactor Safety (December 1972).

5. D. S. Nielsen, "Use of Cause-Consequence Charts in Practical Systems Analysis", *Conference on Reliability and Fault Tree Analysis, September 3-7, 1974, Operations Research Center, Berkeley, Calif.*

6. J. R. Taylor, *A Formalization of Failure Mode Analysis of Control Systems,* Electronics Department, Danish Atomic Energy Commission, Research Establishment RISØ (September 1972).

7. J. R. Taylor, *A Semiautomatic Method for Qualitative Failure Mode Analysis,* RISØ-M-1707, Danish Atomic Energy Commission, Research Establishment RISØ (April 1974).

8. J. R. Taylor, "Sequential Effects in Failure Mode Analysis", *Conference on Reliability and Fault Tree Analysis, September 3-7, 1974, Berkeley Calif.*

9. J. B. Fussell, "Fault Tree Analysis – Concepts and Techniques", NATO Advanced Study Institute on Generic Techniques of System Reliability Assessment, July 1973, Liverpool, England.

10. D. F. Haasl, "Advanced Concepts in Fault Tree Analysis", *System Safety Symposium, June 8-9, 1965, Seattle, The Boeing Company.*

11. W. E. Vesely, "A Time-Dependent Methodology for Fault Tree Evaluation", *Nuclear Engineering and Design, 13,* 2, (August 1970).

12. W. Mendenhall, *Introduction to Probability and Statistics,* Second Edition, Belmont: Wadsworth Publishing Co., Inc., 1967.

13. N. C. Rasmussen, "The AEC Study on the Estimation of Risks to the Public from Potential Accidents in Nuclear Power Plants", *Nuclear Safety, 15,* 4 (July-August 1974).

14. Babcock & Wilcox Co., *1,000 MWe LMFBR Follow-On Study*, BAW-1328, prepared for Argonne National Laboratory (1969).

15. J. D. Esary and H. Ziehms, "Reliability Analysis of Phased Missions", *Conference on Reliability and Fault Tree Analysis, University of California at Berkeley, September 1974.*

16. *OECD Halden Reactor Project Quarterly Progress Report, July to September 1974,* Institut for Atomenergi, P.O. Box 173, Halden, Norway.

17. D. S. Nielsen and B. Runge, "Unreliability of a Standby System with Repair and Imperfect Switching", *IEEE Transactions on Reliability, R-23,* 1 (April 1975).

18. D. S. Nielsen, O. Platz, B. Runge, "A Cause-Consequence Chart of a Redundant Protection System", *IEEE Transactions on Reliability, R-24,* 1 (April 1975).

19. R. A. Evans, "Fault-Trees and Cause-Consequence Charts", *IEEE Transactions on Reliability, R-23,* 1 (April 1974).

DISTRIBUTION RECORD FOR ANCR-1273

External

225 - UC-79h - LMFBR - Structural Materials and Design Engineering (Base Technology)
TID-4500, R64

Internal

1 - Chicago Patent Group - ERDA
9800 South Cass Avenue
Argonne, Illinois 60439

3 - A. T. Morphew, Classification and Technical Information Officer
ERDA-ID
Idaho Falls, Idaho 83401

1 - R. J. Beers, ID

1 - P. E. Litteneker, ID

1 - R. E. Swanson, ID

1 - V. A. Walker, ID

1 - R. E. Wood, ID

29 - Special Internal Distribution

11 - INEL Technical Library

Total Copies Printed - 274