UNCLASSIFIED
(CLASSIFICATION)

DOUGLAS UNITED NUCLEAR, INC.
RICHLAND, WASHINGTON

DOCUMENT NO.
DUN-SA-157

DATE
October 20, 1970

COPY NO.

TITLE

FAULT TREE ANALYSIS WITH PROBABILITY EVALUATION

ISSUING FILE

CONF-701109--6

AUTHOR

P. A. Crosetti

DISTRIBUTION

NAME	BUILDING	AREA	NAME	BUILDING	AREA

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

ROUTE TO	PAYROLL NO.	LOCATION	FILES ROUTE DATE	SIGNATURE AND DATE
Dunn Lab				
DTIE				

UNCLASSIFIED
(CLASSIFICATION)

DUN-SA-157
CONF-701109--6

FAULT TREE ANALYSIS WITH PROBABILITY EVALUATION

SUMMARY

Reliability analysis is playing an increasingly important role in quantitative assessment of system performance for assuring nuclear safety, improving plant performance and plant life, and reducing plant operating costs. In particular, fault tree analysis with probability evaluation provides an all inclusive, versatile mathematical tool for analyzing complex systems. Its application can include a complete plant as well as any of the systems and subsystems. Fault tree analysis provides an objective basis for analyzing system design, performing trade-off studies, analyzing common mode failures, demonstrating compliance with AEC requirements, and justifying system changes or additions. The logic of the approach makes it readily understandable and, therefore, it serves as an effective visibility tool for both engineering and management.

INTRODUCTION

In general, reliability engineering is an applied science which is concerned with utilizing matter, the property of matter, and the physical forces involved, to achieve material having known reliability characteristics. For purposes of this discussion, reliability analysis will primarily involve the quantitative evaluation of plant systems, subsystems, equipment, and components to determine their performance in terms of a specified mission duration when used in the manner and for the purpose intended. This also includes predicting the future performance of systems by quantitatively evaluating their performance on the basis of knowledge of their equipment, components, functions, operating environments, and their inter-relationships.

Fault tree analysis is considered one of the more powerful analytical techniques applied within the aerospace industry to evaluate critical safety hazards. In the nuclear industry, quantitative assessment of system performance can play a key role in assuring nuclear safety, in improving plant performance and plant life, and in reducing plant operating costs.

In recognizing the need for more responsive analytical techniques, Douglas United Nuclear, prime contractor for the Atomic Energy Commission for the operation of the large Hanford nuclear reactors and fuels fabrication facilities, some years ago adapted and applied the fault tree technique to the reactor plants at Hanford. The technique has proven to be a cost-effective systematic, descriptive analysis approach that can be applied to safety and operational analysis of systems from their conception through the design, manufacturing, testing, and operation phases. In particular, fault tree analysis provides analysis flexibility which ranges from equipment analyses to overall plant analyses incorporating all the influencing elements on a total-system basis. The deductive approach used in this technique is particularly useful for evaluating design consistency and reliability, for judging alternatives, for determining acceptability of trade-offs, and for analyzing multiple failure combinations and common mode failures in complex systems. It readily allows analyses to consider the rate at which failures or events are detected after they occur (detection times) and the rates at which they are restored to normal (repair time). Furthermore, phases of operation, such as the reactor shutdown phase, the startup phase, and the normal operation phase, can be considered with provisions for failures to carry over from one phase to the next phase and for failure rates and repair rates to change between phases.

Conventional reliability analyses techniques are inductive in nature and are primarily concerned with assuring that hardware will reliably accomplish its assigned functions. The fault tree method is concerned with assuring that all critical activities are identified and eliminated or controlled. A system, when defined in terms of the all encompassing analysis capability of fault tree analysis, is a composite, at any level of complexity, of operational and support equipment, personnel, facilities, and software which are used together as an entity and are capable of performing and/or supporting an operational role.

DISCUSSION

Fault tree analyses provides a deductive functional development of a specific final undesired event through logic statements of the conditions which could cause the event. Once the final event is defined for assessing system performance, this method provides a concise and orderly description of the various combinations of possible occurrences within the system that could result in the pre-defined event. Since these occurrences are event-oriented, they consider both hardware and nonhardware influences. At the same time, this method provides a systematic means for determining the more significant subsystem and basic input contributions to the probability of causing the event. These failure contributions can then be used to locate and identify major contributors to system failure.

The fault tree can be used as an effective visibility tool and can provide a convenient format for probability evaluation, system analysis, and trade-off study use. The technique requires that only failure-related events be considered and does not require the analysis of failures which have no

effect. Basically, fault tree analysis involves the following steps:

1. Define the undesired event to be used for assessing system performance.

2. Acquire an understanding of the system being evaluated.

3. Analyze the system to determine the higher order functional events which can cause the predefined system fault condition.

4. Continue the fault event analysis to determine the logical inter-relationships of lower order events which can cause them.

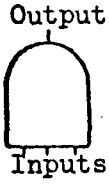
5. Develop a tree of logical relationships among input fault events which are defined in terms of basic, identifiable, independent faults which can be assigned known probability values.

This process results in the functional development of a fault tree using Boolian algebra logic gates to interconnect the events which could cause the specified final event. In other words, the undesired event is the consequence of those basic independent faults which, singularly or in combination, terminate direct paths to the top of the fault tree.

Commonly used fault tree symbols are shown in Figure 1 and Figure 2. The logic symbols (gates) are used to interconnect the events which could cause the specified final event. The logical gates which are most frequently used to develop fault trees include the basic AND and OR logical expressions. The logical "AND" gate provides an output event only if all input events are present simultaneously. The logical "OR" gate provides an output event if one or more of the input events are present. The more frequently used

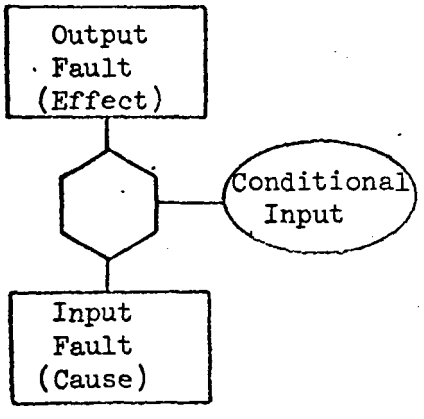
FAULT TREE SYMBOLS - I

LOGIC SYMBOLS

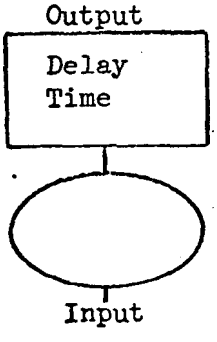


AND Gate
Coexistence of all inputs is required to produce output.

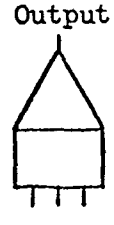
OR Gate
Output will exist if at least one input is present.



INHIBIT Gate
Input produces output directly when conditional input is satisfied.



DELAY Gate
Output occurs after specified delay time has elapsed.

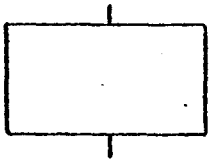


MATRIX Gate
Output is related to one or more unspecified combinations of undeveloped inputs.

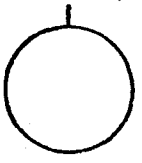
FIGURE 1

FAULT TREE SYMBOLS - II

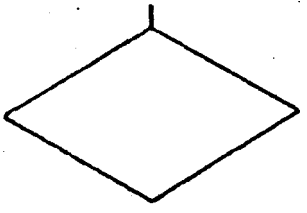
EVENT SYMBOLS



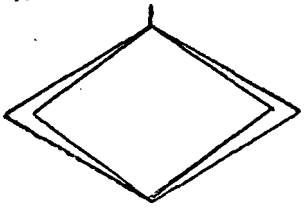
Rectangle
A fault event usually resulting from the combination of more basic faults acting through logic gates.



Circle
A basic component fault - an independent event.

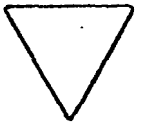


Diamond
A fault event not developed to its cause



Double Diamond
A significant undeveloped fault event that requires further development to complete the fault tree.

Triangle
A connecting or transfer symbol.



Upside Down Triangle
A similarity transfer - the input is similar but not identical to the like identified input.

FIGURE 2

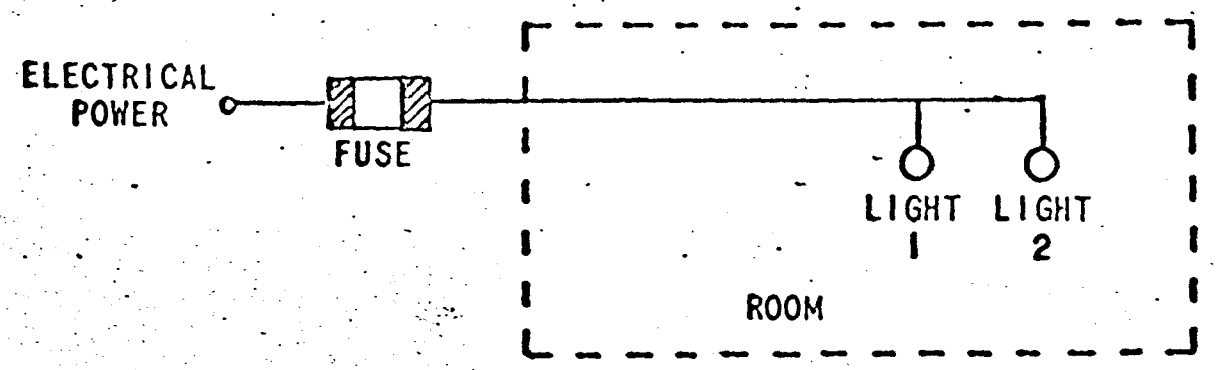
event symbols in fault tree models are the circle and triangles. The circle defines a basic system component or fault input which can be assigned an MTTF (mean-time-to-failure) value and can also be assigned an MTTR (mean-time-to-repair) value, and an MTTD (mean-time-to-detection) value. The triangle indicates a transfer. A line from the side of the triangle (transfer out triangle) denotes an event transfer out from the preceding "logic" gate with the same identification number. A line from the apex of the triangle denotes an event transfer into the succeeding logic gate from the transfer out triangle with the same identification number.

To illustrate basic event symbol usage, an oversimplified fault tree model was prepared to describe a remote, underground room. Required conditions include a minimum level of light intensity and one light bulb provides this level. The room has two light fixtures supplied by one 120 VAC power line. The existing room configuration is shown schematically in Figure 3 and the fault tree for the undesired event, a dark room, is also shown. Figure 4 illustrates the use of input parameters for this case. The input parameters include the MTTF, the MTTD, and the MTTR for each basic input event. Reference 1 discusses this example in more depth including event detection and repair times and trade-off study illustrations.

FAULT TREE PROBABILITY EVALUATION

The fault tree is not only useful as a visibility tool for presenting the various combinations of events in a system, but also as a convenient format for the probability evaluation of the undesired event and all the combinations of events that are most likely to cause the top event.

FAULT TREE ANALYSIS TECHNIQUE



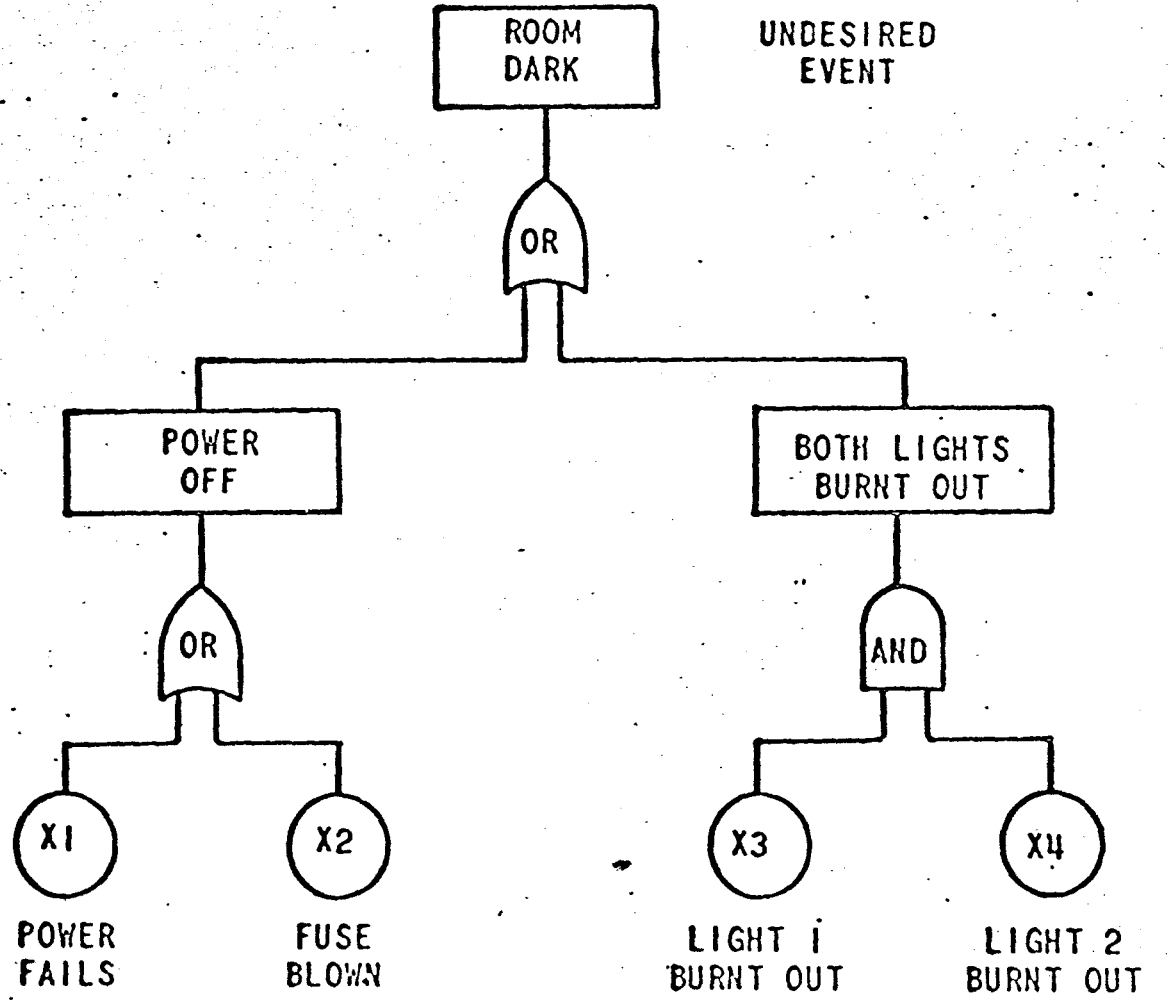
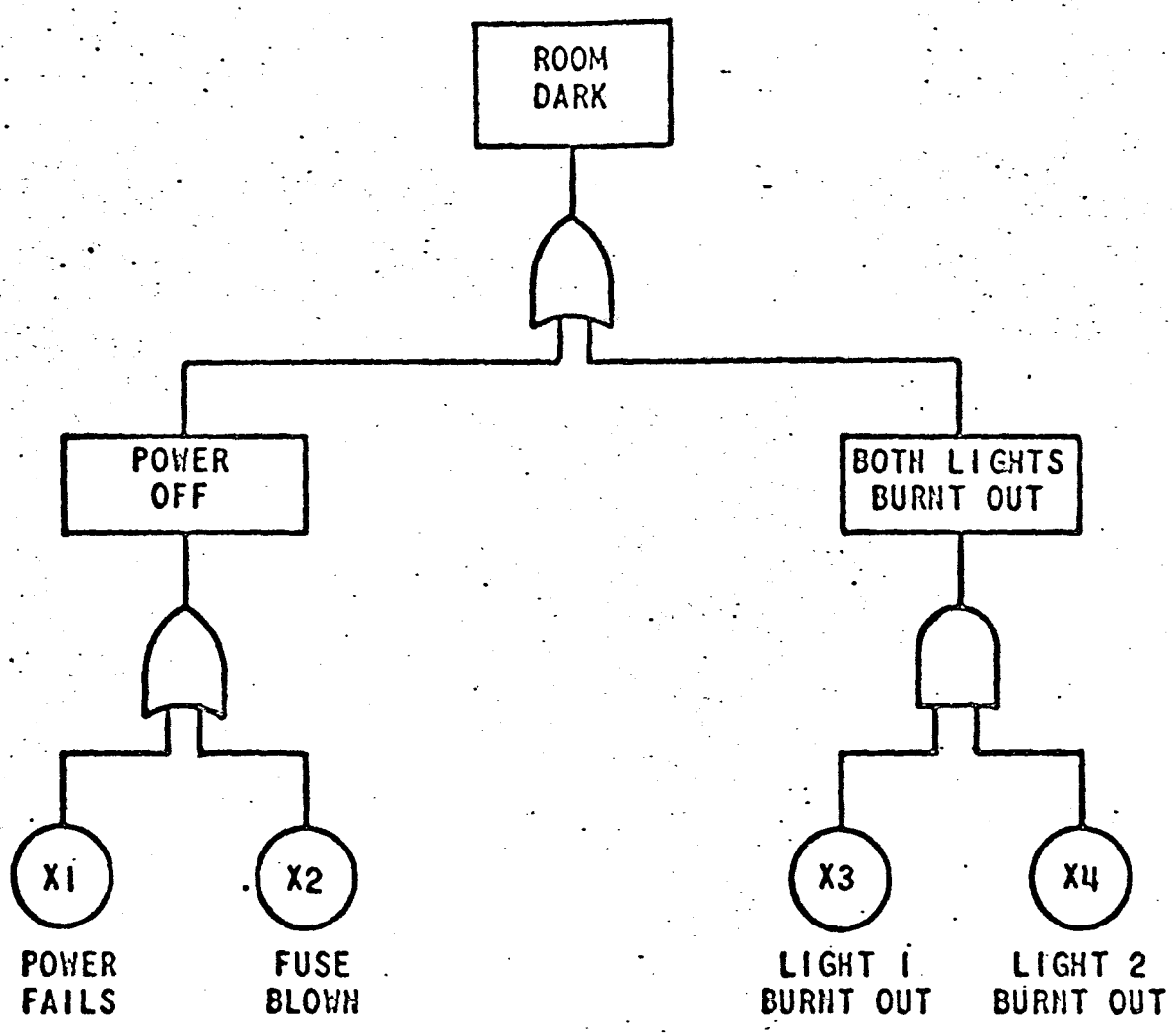


FIGURE 3

EXAMPLE



PARAMETERS

MTTF	100,000	100,000	2,000	2,000
MTTD	0	24	360	360
MTTR	.5	.2	.2	.2

FIGURE 4

Many approaches to directly calculate fault tree probabilities have been attempted. Some of these techniques have proven useful for fault trees representing simple systems involving a small number of events and simple logical relations (usually AND and OR gate logic only). However, they have generally been unsuccessful when applied to fault trees which authentically represent large complex operating systems due to the large number of failure paths and the need to consider phases of system use, subsystem repair and detection conditions which are independent of the basic input events, and other special operational and use conditions.

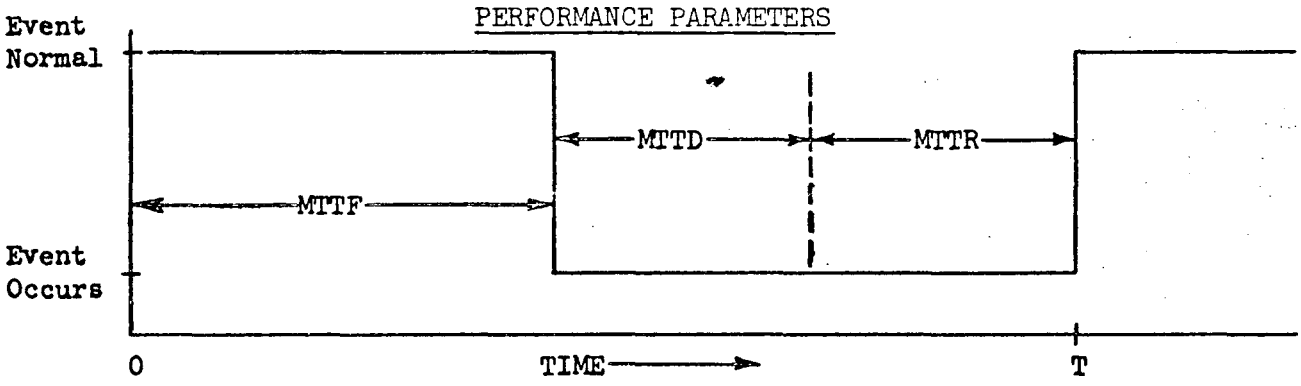
Analytical Calculations

The simple probability method is an analytical technique used to calculate the probability of each combination of events that can cause the top event. The technique can be applied to fault trees which are limited to simple AND and OR logic gates and nonrepairable independent events. The probability of the output event of an AND gate is the product of the probabilities of occurrence of its input events and the probability of the output event of an OR gate is approximately the sum of the probabilities of occurrence of its input events (if the sum of the probabilities is much less than 1.0).

The probability information can then be used to rank the various paths and to calculate the total probability for the top event. However, for large systems, there are too many failure paths to permit calculation of the probability for every path. Fault trees involving 200 events will normally have thousands of paths to the top event. A 400-event tree will have millions of paths. Also, fault trees usually involve gates having dependent inputs. These occur because of the structure of the fault tree since the basic input events are independent.

Input dependencies can be removed by using the Boolean substitution and reduction technique to provide a simplified fault tree for calculation purposes. However, the technique does not always put a logic diagram into a completely dependent free form, but it does result in a redundant free form. That is, the probability for a given event appears only once in each term of the expanded probability expression and the same combination of events will not be added more than once. Also, the failure to remove all dependencies does not introduce a significant error if the probabilities are small. However, the simplification of the fault tree does alter the original tree structure. This must be considered when the original tree structure is desired for visibility or path evaluation use.

Since equipment is often monitored for failure, the consideration of repair is essential since detection of and correction of failures serves as a very powerful performance tool. If repair was not considered, large errors can occur in the probability evaluation results and there would be no way to evaluate the effectiveness of various maintenance programs and the monitoring techniques. The analytic method for repair, commonly called the "lambda-tau" method, extends the original fault tree capability. This provision for repairable events is restricted to duration times for basic input events. The outputs of gates are assumed to remain for as long as the gate input events are in effect. The event performance is illustrated below. The event occurrence rate can be thought of as being analogous



to a component mean time to failure and the event duration time as being analogous to the sum of the component mean time to detection and the component mean time to repair.

The method requires a redundant free expression of the logic diagram and incorporates several restrictive approximations. To obtain redundant free expressions, the technique is usually used in conjunction with the Boolean substitution and reduction technique. The approximations are good if the product of the input event failure rate times the mission period time is less than one and if the event repair time is less than the mission period time. Furthermore, only standard AND and OR logic gates can be treated. These analytical techniques are discussed in more depth in reference 2.

Monte Carlo Simulation

A feasible approach to probabilistic evaluation of event logic diagrams is to concentrate the effort on the dominant paths. This can be accomplished using Monte Carlo simulation, the simulation being performed on a computer using an event logic simulation program. Probability data are inputted and the simulation program represents the fault tree on a computer to provide quantitative results. In this manner, thousands or millions of trial years of performance can be simulated. A typical simulation program involves the following steps:

1. Assign failure rate data to input fault events within the tree, descriptive mission data, and if desired, repair rate data.
2. Represent the fault tree on a computer to provide quantitative results for the overall system performance, subsystem performance, and the basic input event performance. These results can include

the specified final event probability of failure and success, total failure information, availability, and downtime results.

3. List the failures which lead to the undesired event and identify critical path contributing event results.
4. Compute and rank basic input failure and availability performance results.

In accomplishing these steps, the computer program simulates the fault tree and, using the input data, randomly selects rate data from assigned statistical distribution parameters, and then tests whether or not the specified final event occurred within the specified time period. Each test is a trial, and a sufficient number of trials are run until the desired quantitative resolution is obtained. Each time the final event occurs, the contributing effects of input events and the logical gates (paths) in causing the specified final event are stored and listed as computer output. The resultant output provides a detailed perspective of the system under simulated operating conditions and provides a quantitative basis to support objective decisions. Engineering, operational and maintenance alternatives can be readily evaluated and sensitivity analyses can be performed by varying input data over pre-selected ranges.

Scaling

The simulation technique, as described above, is commonly called Direct Simulation. However, fault tree diagrams usually represent improbable events and the number of trails required to obtain the desired quantitative resolution can become prohibitive.

In order to reduce the computer run time to an acceptable level, a scaling technique is often used. As applied to fault tree simulation, the purpose

of scaling is to generate events in a manner which increases the frequency with which the various event combinations occur, while retaining the feature that dominant paths occur most frequently and that path and event ranking is not disturbed. The method of scaling usually involves increasing the frequency of occurrence of the basin input events and extending the duration time (time from event occurrence to the time the event is restored to normal) of the input events once they have occurred. Both non-linear and linear scaling techniques are used. The actual improvement in computer time is dependent upon the fault tree structure, the simulation program features, and the scaling technique used.

An example of an early fault tree simulation program which uses a linear scaling technique is discussed in reference 3. Although other scaling techniques[2] and many simulation program features exist, this example does illustrate a typical approach used in computer simulation. It also provides a sample of the fault tree features which can be accommodated and of the computer output which can be obtained from simulation programs.

Phasing

If systems are relatively static from an analysis standpoint, it may not be necessary to consider mission phasing for either logic structure or event occurrence and duration data. Programs without mission phasing are called single phase programs. However, many systems are phase dependent. That is, the logic structure or event occurrence and duration data will change between phases of operation. Typical nuclear plant phases may include reactor shutdown, startup, and operating phases. Programs with mission phasing are called multiple phase programs.

Data

Typical sources of data used for fault tree probability evaluation include data banks, industrial data, performance data, and manufacturer's data. In addition, systems and equipment performance data acquired from the Hanford reactors are readily obtained through a computerized data storage and analysis program.[4,5,6] Importantly, techniques such as fault tree analyses with probability evaluation allow the use of sensitivity analysis to provide a means of evaluating the significance of events for which data do not exist or are of poor quality. Then events which are not significant can be eliminated, and the significant events can be reviewed in terms of the data quality needed or the degree of control required.

Related Analyses

There are other qualitative and quantitative analyses which can be performed in conjunction with or used in support of fault tree analysis with probability evaluation and in some cases can include the fault tree method. Typical of these methods are the following aerospace industry analysis:

Gross Hazard Analysis

Preliminary Hazard Analysis

Operating Hazard Analysis

Failure Rate Analysis

Subsystem Hazard Analysis

Fault Hazard Analysis

Failure Mode and Effects Analysis

Failure Mode Effects and Criticality Analysis

The usefulness of these analyses is determined by the type and depth of the analysis being performed and cost and time constraints. However,

analysis, such as fault hazard analysis, failure mode and effects analysis, and failure mode effects and criticality analysis, can be particularly useful in helping assure that all important hardware related events have been considered during the fault tree analysis.

Fault Tree Analysis Application by Douglas United Nuclear

The System Effectiveness Program developed by Douglas United Nuclear provides the means for measuring, and the basis for optimizing, the performance of nuclear reactor systems. The analyses serve to enhance safety reliability, operating reliability, and cost-effectiveness through the use of equipment performance information and Fault-Tree Analysis techniques.

The nuclear safety analysis phase evaluates the probability that the reactor safety systems (those systems designed to minimize the nuclear consequence of a plant incident) will function as required, when required, and for the required period of time.

In assessing nuclear safety performance, possible reactor plant incidents that may require the safety system to function are first defined. A system model is then prepared for each system selected for analysis using Fault-Tree Analysis techniques. The model and related occurrence (failure) and duration (repair) data are then arranged in format for computer processing. The computer programs, which are Monte Carlo simulation programs, are then used for quantifying system performance results and for providing a ranking of the events most likely to cause total system failure. To optimize the system performance, trade-off studies are represented by model changes and the performance results are determined to permit the selecting of the optimum arrangement from a systems effectiveness standpoint.

The reactor operating performance phase evaluates the probability that the reactors will achieve an uninterrupted scheduled operating period, or that, once started, the reactors will not be shutdown for other than a scheduled outage. The objective of this phase of the program, therefore, is to increase plant availability by identifying performance limiting systems, initiating optimization action, and measuring the results of the improvement action.

REFERENCES

1. P. A. Crosetti and R. A. Bruce, "Commercial Application of Fault Tree Analysis," Ninth Reliability and Maintainability Conference, Annals of Reliability and Maintainability - 1970, Volume 9, Page 230.

2. R. J. Schroder, "Fault Trees for Reliability Analysis." Paper presented at the 1970 Annual Symposium on Reliability, Los Angeles, January 1970.

3. P. A. Crosetti, "Computer Program for Fault Tree Analysis," April 1969, DUN-5508. Available through Clearinghouse for Federal Scientific and Technical Information, Springfield, Va. 22151.

4. P. A. Crosetti and M. L. Faught, "Systems Approach to Nuclear Plant Protective Systems Data Program," DUN-SA-154, October 13, 1970. Paper presented at IEEE 1970 Nuclear Power Systems Symposium, November 1970.

5. G. E. Greger, D. A. Snyder, and P. D. Gross, "Description and Uses of a Critical Systems Data File for Nuclear Plants," The American Society of Mechanical Engineers, United Engineering Center, 345 East 47th Street, New York, New York 10017.

6. T. M. Clement, "Nuclear Equipment Maintenance Standards Enhance Reactor Reliability," Power (April 1968), Page 90.