NUCLEAR ENGINEERING

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

PHYSICAL DEPENDENCIES IN ACCIDENT SEQUENCE ANALYSIS

by

N. Siu, C. Acosta, and N. Rasmussen

October, 1989

MITNE–288

Nuclear Engineering Department
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

PHYSICAL DEPENDENCIES IN ACCIDENT SEQUENCE ANALYSIS

by

N. Siu, C. Acosta, and N. Rasmussen

October, 1989

MITNE–288

Prepared for:

Office of Nuclear Regulatory Research
United States Nuclear Regulatory Commission
Washington, D. C. 20555

Project Monitor:

Thomas G. Ryan

Grant Number NRC–04–88–143

First Progress Report
Physical Dependencies in Accident Sequence Analysis
N. Siu, C. Acosta, N. Rasmussen

1. INTRODUCTION AND SUMMARY

This progress report highlights the work over the period 5/2/88 to 6/30/89 under grant NRC–04–88–143, "Physical Dependencies in Accident Sequence Analysis" [1]. The objective of this project is to develop and demonstrate an improved accident sequence analysis methodology which will allow better treatment of the risk associated with plant dynamic behavior, including the effects of "physical dependencies" between plant safety systems.

In order to accurately describe plant risk, a risk assessment study must identify and quantify all significant dependent failures of plant safety systems. As described in Ref. 1, conventional event tree/fault tree approaches are unable to perform either task in many cases where an initial failure increases the likelihood of subsequent failures because of the response of plant process variables (e.g., pressure and temperature) to the initial failure. In such cases, the failures are probabilistically dependent; the term "physical dependencies" is used to refer to the coupling between failure events due to the behavior of plant process variables (as opposed to "logical dependencies" caused by the direct interaction of systems).

The original objectives of the project, as stated in Ref. 1, were to develop and demonstrate an improved accident sequence analysis methodology which would allow better treatment of physical dependencies. The project approach intended to accomplish this objective consisted of five tasks:

1) Identification of actually experienced nuclear power plant accident scenarios in which physical dependencies have been significant.
2) Identification of candidate models for efficiently treating these scenarios.
3) Comparison of candidate models in case studies.
4) Characterization of models for application to general scenarios.
5) Documentation and dissemination of results.

Based upon the research performed to date, slight changes in the project objectives and in the task definitions have been made. The project objective, mentioned at the beginning of this section, now addresses plant dynamic behavior. This broadening of the original objective allows more careful treatment of the interface between the accident sequence model and operator models. The current

Task 4 now emphasizes to a greater degree the implementation of a particular modeling framework (the DYLAM methodology). Less effort will be spent on identifying and characterizing a wide variety of candidate methodologies for treating process variables (Tasks 2 and 3).

The motivation underlying these changes in the project objective and tasks is discussed in Section 2 of this report. The remainder of the report discusses the results obtained to date.

Section 3 summarizes the results obtained for Task 1. It is shown that relatively few of the accident and accident precursor scenarios experienced in the years 1969–1979 and 1985 appear to involve physical dependencies. This indicates that the current event tree/fault tree approach currently used in conventional risk studies is probably appropriate for the scenarios likely to be observed in power plants. However, physical dependencies have played significant roles in at least one key incident: the TMI–2 accident (3/79). This strengthens the belief that proper treatment of physical dependencies is required to assure identification of a plant's most risk–significant scenarios.

These conclusions cannot be stated more strongly because the summary information used (Refs. 2 and 3) is not tailored to identify physical dependencies. Later in this project, additional research on a number of salient accidents will be performed to make the conclusions more definitive.

Section 4 of this report discusses results from Tasks 2 and 3. Four general methodologies for treating physical behavior are investigated: expanded event trees, digraphs, Markov models, and simulation models. The first three approaches, while easy to apply to simple situations, rapidly become unmanageable when dealing with complex power plant behavior during a transient. The problem is that they are not really tailored to treat dynamic scenarios, being essentially static models (the model structure cannot change with time). These three approaches can be modified to incorporate some level of dynamics, but the resulting models are often too complex to be understood by others than the analysts responsible for the models. On the other hand, the simulation approach employs an explicitly time–dependent representation of the system being treated. Rules of arbitrary complexity governing the evolution of a scenario can be incorporated without modification to the basic structure of the approach. The problems associated with simulation (e.g., sampling to include rare events, analysis of results from the simulation "black box," and the potential need for large amounts of computer time) appear to be easier to deal with than the problems associated with the other approaches.

The general idea of simulation can be applied to the accident scenario problem in many ways. Section 4 investigates the DYLAM (Dynamic Logical Analysis Methodology) approach, a physical equation–based method that has been proposed for accident sequence analysis [4]. This approach appears to be especially promising and will be further investigated in this project.

2. CHANGES TO PROJECT OBJECTIVES AND TASK STRUCTURE

As stated in Ref. 1, the original proposal to the NRC for this project emphasized the development of a methodology for treating physical dependencies between plant systems. Although Ref. 1 recognized that human actions could be significant contributors to this class of dependent failures, and that models for these actions would be needed in order to use the methodology for quantitative analysis, the proposed research did not emphasize the development of explicit methods to integrate operator models into the overall analysis. It was felt that physical dependencies could be treated largely by broadening the definition of the current plant state to include the current values of process variables; as in a conventional event tree analysis, the impact of the plant state on operator actions would then be treated by the analysts performing the operator actions analysis.

Research conducted as part of this project and as part of a parallel project on operator crew modeling (recently funded by the NRC [5]) now indicates that the objective of this project needs to be broadened slightly. If operators are not included in the analysis, the earlier model, in which the likelihood of system failure is modeled as being only dependent on the current values of process variables and component states, appears to be reasonable. However, when operator actions are included, the likelihood of failures is also affected by each operator's state of mind. Note that this seems to be especially important when considering strings of failures. The operator's current state of mind, in turn, is likely to be affected by the entire set of preceding events, i.e., the scenario history. Thus, factors not modeled in current event trees, including the exact order and timing of failure events, and the time–variation of process variables, may have an impact on the likelihood of multiple failures.

The above discussion points to the need for a model that treats the dynamic evolution of an accident scenario. This model will still predict the current values of process variables, as in the original proposal. In addition, however, it will now keep track of the scenario history, i.e., how the current state was reached. Moreover, much greater attention will be paid to the interface with the operator model (the latter determines what kind of information needs to be carried in the accident sequence model). The interaction between this project and the parallel MIT operator crew model project [5] will ensure the integration of the two models.

The broadening of the intended project output (from a model to treat physical dependencies to one that models plant dynamic behavior) has implications for the task structure originally proposed in Ref. 1 and summarized in Section 1. In particular, Tasks 2–4 envisioned a detailed comparison of a number of competing methodologies, with the distinct possibility that no one approach would be uniformly superior to the others. It now appears that logically–oriented methodologies (event trees/fault trees and digraphs) are inappropriate because of their weakness in handling time dependence. Rather, a simulation–oriented approach, which not only handles time dependence naturally, it also can keep track of scenario history easily, seems to be clearly preferable.

One particular simulation approach, the DYLAM approach, in which time–dependent event tree structures are generated dynamically, promises to be especially useful and will be employed in this project. The technical details of the approach are described in Section 4. Regarding the impact on the project task structure, Tasks 2 and 3 can be regarded as being essentially complete, even though the candidate models discussed in Section 4 of this report were not actually applied in a realistic analysis. On the other hand, Task 4 will be greatly enlarged, as follows.

DYLAM was originally proposed as a methodology for systems analysis. As a methodology for accident sequence analysis, it has a severe drawback in that the number of sequences to be analyzed can become unacceptably large very quickly. Therefore, it has not, to our knowledge, been applied in a detailed nuclear plant accident sequence analysis (where dependent failures are of central importance). The methods proposed in Ref. 4 to limit the number of sequences (truncation based on probability or on size of cut sets) are not believed to be sufficient to allow practical implementation of the methodology. Therefore, one of the objectives of Task 4 will be to develop methods to allow practical implementation of DYLAM.

A second problem with using DYLAM in an accident sequence analysis is that the model must keep track of the full history of a scenario. Thus, in addition to the sequence and timing of system failures, and the process variable history, it must also keep track of each operator's state of mind (indexed in some fashion). Methods for doing this will also be investigated as part of Task 4.

3. ACCIDENT SCENARIO IDENTIFICATION

The purpose of this task is better determine the significance of the issue of physical dependencies (raised in Ref. 1), and, to characterize how these dependencies might come about. Events A and B are said to be dependent if $P\{A \text{ AND } B\} \neq P\{A\} \cdot P\{B\}$. "Physical dependencies" exist if the performance of a system causes a change in magnitude of at least one process variable (e.g., coolant temperature or coolant pressure), which in turn, directly or indirectly influences the operation of another system.

To perform this task, a large group of the accident and accident precursor scenarios actually experienced by U.S. commercial nuclear plants is investigated. Scenarios involving physical dependencies are identified and characterized. No effort is made to perform a formal statistical analysis; the emphasis is to develop a representative set of different scenarios that current models should address.

3.1 Data Base for Scenarios

The references used in the review of observed scenarios are the 1985 and 1969–79 precursor studies [2,3]. These studies are reviews of the licensee event reports (LERs) for incidents that occurred during the indicated years at U.S. commercial light water reactors (LWRs) in order to identify and classify precursor events that could lead to severe core damage accidents. When accompanied with postulated events, these precursors, which involve either initiating events or system (or equipment) failure, could result in an inadequate core cooling situation. The 1969–79 precursor study initially screened 19400 LERs, reviewed in detail 500 LERs, and eventually identified 169 precursors. The 1985 precursor study, on the other hand, started with 3000 LERs, of which 1400 received detailed review, and classified 63 events as precursors to severe core accidents.

The events that were identified in Refs. 2 and 3 as being precursors to severe core accidents involved the failure of at least one needed mitigating system, the degradation of more than one needed mitigating system, or an actual initiating event that required safety system response. A mitigating or safety system refers to a system designed to prevent an abnormal occurrence from causing safety–related effects and to bring the plant to an innocuous condition. An initiating event event, on the other hand, is an occurrence that perturbs the operation of the plant and requires the performance of plant systems for mitigations. An initiating event could be a system failure or an external hazard.

3.2 Selection of a Representative Set of Scenarios

Although a large number of scenarios are covered by the precursor studies, not all of these scenarios are relevant to this study. To efficiently identify important scenarios, two criteria are employed to screen the list of precursors. These criteria should be applicable in screening other precursor studies as well. They are:

1. Events must have occurred while the plant was operational. This ensures that the physical dependencies identified are relevant during plant operation, but can omit some important scenarios.
2. Events of interest must have involved at least two safety systems or functions (since plant–level dependencies involve at least two systems). This is implemented by considering only precursor events whose event trees have at least two affected headings or branches.

Application of the above criteria to the 1969–1979 precursors reduces the initial 169 precursors to 27 (Table 1). Of these 27 precursors, only one appears to have involved a physical dependency. This event is the well–known Three–Mile Island 2 (TMI–2) accident which occurred on March 28, 1979. During the early part of the accident, the failure of the pilot operated relief valve (PORV) to reclose resulted in the rise of the pressurizer level, which in turn, led the operator to throttle the high pressure injection (HPI) system. In this event the two interacting "systems" were the PORV and the HPI. The water level was the involved process variable.

To model this sequence, the current event tree approach can show that the PORV opens, the PORV fails to close, and the HPI fails. The water level–operator interaction, however, is not shown. For example, Figure 1 shows the TMI–2 sequence using a modified transient event tree from the Oconee risk study [6]. An analyst determining $P\{\overline{U}|T\overline{BQ}\}$ is supposed to recognize the interaction and incorporate an appropriate human error term in the fault tree for the HPIS.

Applying the criteria to the 63 precursors in the 1985 precursor study results in 24 candidate events (Table 2), of which four appear to have involved physical dependencies. These incidents occurred at Davis–Besse 1 (9 June 1985), Catawba 1 (22 June 1985), Hatch 1 (15 May 1985), and Rancho Seco (26 December 1985). Davis–Besse 1, Catawba 1 and Hatch 1 involved the multiple opening and reclosing of relief valves. The affected functions in these incidents were the demand for the relief valves to open and the subsequent demand for the relief valves to reclose. Pressure was the involved process variable. Standard event tree models for multiple opening and reclosing of relief valves treat the recurring functions (opening and reclosing of the relief valves) just once in the tree. It is possible for the analyst to conservatively quantify the branching probability (e.g., by using the probability of failure for multiple trials). However, there are no guidelines as to the degree of conservatism to be used. Further, it is not clear that the remainder of the sequence is independent of the number of relief valve open/close cycles (or of the cause for the differing number of cycles). For example, the operators' view of the plant and their subsequent actions may depend on the number of cycles observed.

During the Hatch 1 and Rancho Seco incidents, potential failure events involving physical dependency occurred. In both cases, equipment failure led to rapid cooldown, exposing the reactor vessel to failure. In the Hatch 1 incident, the failure of the safety relief valve (SRV) to reclose for approximately 34 minutes which resulted in the temperature cooldown rate that exceeded the technical specifications. In the Rancho Seco incident, a stuck open auxiliary feedwater (AFW) manual isolation valve erroneously allowed water to overfill a steam generator and prevented the early recovery from an overcooling condition. Thus, a rapid reactor coolant system (RCS) cooldown and repressurization ensued, causing the RCS to enter the B&W–designated pressurized thermal shock (PTS) region. In both instances (Hatch 1 and Rancho Seco) the RCS temperature was an involved process variable and the integrity of the vessel was the affected second function. The first safety function was the safety relief valve for the Hatch 1 incident and the AFW system for Rancho Seco. This physical behaviour is generally treated in recent event sequence analyses, as shown by the Event Sequence Diagrams for the Seabrook study [7]. However, further study is needed to determine if the details of these scenarios are explicitly treated in current analyses.

Based on the current results of the scenario search, two observations can be made. The first observation is that although the 1969–1979 study has more scenarios than the 1985 study, there are more scenarios that displayed physical dependencies in 1985 than in 1969–1979. This could be partially explained by the more detailed reporting in the later report. The second observation is that the data base appears to contain only few scenarios involving physical dependencies. This could be attributed to two factors. First, the actual occurrence rate of such scenarios appears to be low (the occurrence rate of any scenarios involving multiple failures, of which dependent failure scenarios are a subclass, is low), Second, the description of events in the Precursor studies (especially the 1969–1979 study) may not be detailed enough to determine all events that involved physical dependencies (these studies were not designed to identify this type of dependency between failure events).

3.3 Characterization of Scenarios

The physical dependencies identified in the previous section can be categorized into three classes: direct physical dependencies, indirect physical dependencies, and cyclical physical dependencies.

3.3.1 Direct Physical Dependency

Direct physical dependency entails direct interaction of two safety systems (or functions) through process variables. The failure of one system alters the magnitude of the involved process variable, which in turn, influences the performance of another system. Figure 2 depicts this relationship. The process variable either attains a value high enough to cause the second system to attempt to change state or changes its magnitude at a rate faster than the considered safe limit (e.g., excessive cooldown). The use of a set point value, which, if attained by a specified process variable, signals a designated system to change its state, is associated with failure modes of components (e.g., the PORV) in the current PRA approach. The time rate of change of process variable magnitudes, however, are not usually explicitly incorporated.

Two of the scenarios found displayed direct physical dependency. They are the Hatch 1 (May 15, 1985) and the Rancho Seco (December 26, 1985) incidents. As mentioned earlier, the systems (or functions) involved are the AFW and the integrity of the pressure vessel for Rancho Seco and an SRV and the integrity of the vessel for Hatch 1. The process variables are RCS temperature and pressure for Rancho Seco, and RCS temperature for Hatch 1.

3.3.2 Indirect Physical Dependency

Indirect physical dependency requires an intermediary between the demanded systems (Figure 3). The change in the process variable value, caused by the performance of one system, leads the intermediate system to influence the state or response of another system. Either the operators or a control system can act as the intermediate participant. The TMI–2 accident stands out in this category. The first and second failure events in this case were the failure of a PORV to reclose and the failure of the HPI system. Pressurizer water level was the process variable and the operators acted as the intermediate system.

3.3.3 Cyclical Physical Dependency

Cyclical physical dependency involves the multiple occurrence of certain events (Figure 4). An example of this is the multiple opening and closing of relief valves experienced at Davis–Besse 1 (June 9, 1985), Catawba 1 (June 22, 1985), and Hatch 1 (May 15, 1985). The opening and reclosing of the relief valves were the affected functions (these are often separate top events in an event tree). The involved process variable was the RCS pressure. Note that risk studies must often accomodate cyclical events in a static framework using arbitrary assumptions (e.g., the assumption of a limited number of cycles before failure when analyzing HPCI/RCIC systems in BWR PRAs).

4. SURVEY OF CANDIDATE METHODOLOGIES

As discussed in Ref. 1, there are a number of methodologies discussed in the literature that may be useful for incorporating plant physical behavior into accident sequence analysis. The purposes of Tasks 2 and 3 of this study are to identify specific implementations of these methodologies, and to characterize them in order to determine their usefulness.

4.1 Desireable Attributes of Methodology

The above discussions on scenarios involving physical dependencies and some limitations of the current event tree method in treating the scenarios naturally suggest that an improved methodology should explicitly treat process variables. Three important additional requirements are:

1. It must be dynamic, i.e., it must be time dependent. For example, it must accomodate complex behavior introduced by control laws, feedback loops, random timing of events, and the history of the sequence.
2. The accident sequence model created with the methodology must be understandable to reviewers not involved in the construction of the model.
3. The results of the model must also be readily understandable.

Satisfaction of the first requirement will allow clearer integration of models for plant operators and control systems (as compared with the current event tree methodology, where a single logical sequence may represent a wide variety of actual sequences). The second requirement recognizes that the general ideas underlying a particular event tree model are easy to understand (if not the details); an improved methodology should strive for the same level of clarity, even though its details may be more complicated. The third requirement also recognizes a strength of event trees: their results are given in terms of easily understood and ranked scenarios. This is a desirable goal and should be pursued by an improved method (to the extent that it doesn't hide significant contributors to risk).

Four candidate approaches are: expanded event tree modeling, digraph modeling, Markov modeling, and the Dynamic Logical Analytical Methodology (DYLAM). Three of these four approaches will be discussed in the context of a simple system, described below. The digraph modeling discussion is based on a recently completed work which deals with accident sequence analysis [9].

4.2 Description of the Example System

To better illustrate the characteristics of the candidate modeling approaches, we employ an example system adopted from Ref. 9 (see Figure 5). This is a water storage tank system with three control units (two pumps and one valve) and one process variable (the water level).

During normal operation the water level is within a predefined nominal control interval. Any control unit failure causes a system disturbance which could lead to the water level moving out of the nominal interval, either above or below the acceptable interval. Any unit failure and subsequent water level movement out of the interval, however, does not necessarily result in system failure. The water level can be brought back to the nominal interval or be kept within the still–considered success region by the actions of the remaining operational control units.

The system fails when either a dryout or an overflow condition occurs. Dryout happens when the water level is below the lower bound of the acceptable interval; overflow occurs when the water level is higher than the upper bound. Both failure states are considered absorbing states, i.e., the water level stays within these intervals once it gets there. Control rules, which are dependent on the water level, specify the actions of the control units as shown in Table 3.

Relevant assumptions on the properties of the system, useful for the purposes of this demonstration, are:

1. Control units are either on or off.
2. Unit failure rates are constants.
3. Failed units are not repaired.
4. Unit failures are mutually statistically independent. There is no sharing of common elements and consideration of common cause failure.
5. The operational state of a control unit at any particular time depends solely on the magnitude of the water level (Table 3).
6. Control units respond with negligible time delay.

4.3 Description, Application and Analysis of Candidate Methodologies

4.3.1 Expanded Event Trees

The event tree methodology, currently used as the standard approach for accident sequence analysis in nuclear power plant risk studies, is well–documented (e.g., [10–11]). This method represents the evolution of an accident as a sequence of the states of the event tree "top events" (which can include safety systems or operator actions) from an initiating event to some plant damage state. Plant physical behavior enters in determining the consequences of each sequence (which leads to the definition of plant damage states) and in determining the success criteria for the safety systems (e.g., "auxiliary feedwater must be available for 30 minutes").

Although applicable to many scenarios, the event tree method does not handle certain types of scenarios involving physical dependency. One method to improve event trees is simply to increase its level of detail by defining more top events, i.e., to expand the tree. A study of Babcock and Wilcox (B&W) plants resulted in a substantially expanded LOFW event tree [12], containing 21 top events (excluding the initiating event) (Figure 6). The tree treats operator actions and functional headings in considerable detail. Expanded event trees have also been used in human reliability studies. Heslinga [13] used repetition of trees to account for recovery of a faulty action (Figure 7). Apostolakis and Chu, in their analysis of the TMI–2 accident, allow transition among branches corresponding to different headings to account for time–dependent human action [14]. NUREG–1150 applies event tree expansion extensively in its containment analyses [15]. The number of headings, some of which are in terms of process variables, in these event trees ranges from 49 to 107. The tree headings can have more than two branches and can be repeated at different stages of the scenario to account for time dependence. Appendix A shows a list of questions which served as basis for the Surry containment event tree headings.

Some typical modifications of event tree top events that could allow treatment of process variables in an accident sequence analysis are:

a. Inclusion of physical phenomena or process variables as headings in the event trees (e.g. RCS pressure > 2200 psig).

b. Further division or splitting of tree headings. This can facilitate the inclusion of process variables as headings and account for relative timing of events.

c. Employment of more than the normal two branches under a heading.

d. Repetition of some of the headings. This could account for recurring sequences and monitoring values of process variables.

To illustrate how the expanded event tree method would be used to handle the hypothetical example system, the initiating event "pump 1 fails off" is considered. The analysis assumes that the valve and pump 1 have equal flow rates and that the pump 2 flow rate is half that of either pump 1 or the valve.

Figure 8 shows a state transition diagram for the system following the initiating event "Pump 1 fails stopped." This diagram shows all of the possible events that can follow the initiating event. Although this diagram is essentially an event tree, it is somewhat more general than the trees typically used in PRAs. Event headings are placed at each node (in boxes). Each box represents a "snapshot". For example, Box 4 portrays the conditions: the water level is within the nominal control region (but below its initial value), the valve is "good and open," Pump 1 is "failed and stopped," and Pump 2 is "failed and running."

This representation allows incorporation of variable event timing (including different orderings of events) and the explicit treatment of process variables. Note that two calculations must be performed for each transition: the deterministic calculation to determine changes in process variables, and the probabilistic calculation to determine possible hardware state changes (see Appendix B for more details on the latter). Each train of boxes, starting from the first box, constitutes an accident sequence. Hence, the train of boxes 0–1–3–9–17–24–29 is the sequence starting with the failure (off) of Pump 1 (Box 1), the failure (off) of the valve (Box 3), and the failure (on) of Pump 2 (Box 9). When the water level rises above the upper setpoint $\alpha 2$ (Box 24), the condition of Pump 1 changes from "failed and stopped" to "failed and running," as prescribed by the assumptions given in Section 3.2 and by Tables 3 and 5. The sequence eventually ends in an overflow state (Box 29).

The somewhat non–intuitive rules leading to Pump 1 changing states (from stopped to running) while failed result from the modeling simplifications applied in Ref. 9. That model concentrates on the comparision of the current state of a control unit with its desired state; the manner by which the current state was reached is not addressed. This reduces the amount of information that must be carried in the analysis. This simplification, it should be noted, is not required by any of the methods described in this report (including that of Ref. 9). Note also that if the failure is assumed to be associated with the controls for the pump, rather than with the pump itself, the rules of behavior may be rationalized somewhat.

Figure 9 presents a portion of a more conventional representation of the same event sequences shown in Figure 8, being an expanded event tree based on the first four boxes of Figure 8. Although Figure 9 has the benefit of being in a familiar format, it is much less compact than Figure 8. Note that the expanded tree can become extremely large when accounting for different orderings of failure events in different sequences. Furthermore, the expanded event tree does not convey the image of competing processes. For example, the notion that the first transition to occur following the initiating event is the outcome of three parallel random processes (water level dropping towards α_1, valve failure, Pump 2 failure) is much clearer in Figure 8 than in Figure 9.

Thus, in the case of a single process variable, it appears that the general tree diagram of Figure 8 is superior to the conventional expanded event tree of Figure 9. However, both are probably too complicated to employ for situations requiring the explicit treatment of more than one process variable or multiple component states. Even a simple increase in the number of components analyzed increases the complexity of both representations tremendously.

This discussion shows how event trees can explicitly handle process variables through expansion (assuming that there is a process variable simulator to calculate the system physics and update the process variable values). This method is easy in principle and allows decomposition of the tree into smaller and more manageable parts. Human actions can be incorporated in expanded event trees in the same manner as used for conventional event trees.

On the other hand, the methodology is still inherently static. Loops or even simple events at multiple points in time cannot be treated without some arbitrary truncation rules. Furthermore, the treatment of several process variables at the same time can lead to even greater complexity. All of these problems will lead to extremely (and probably excessively) large trees when dealing with realistic accident sequence models. Thus, "conventional" static expanded event trees appear to be impractical for realistic applications. However, if they are modified to treat time explicitly, they can be much more promising. This modification is at the heart of the DYLAM methodology, discussed in Section 4.3.4.

4.3.2 Digraphs

One of the problems mentioned in the case of expanded event trees is that the tree cannot handle loops without some arbitrary truncation rule. This problem is also discussed in Ref. 14. An alternate graphical approach, which allows the correct treatment of loops, involves the use of directed graphs (or digraphs).

Digraphs consist of nodes and directed arcs between the nodes. The nodes can represent different system states (similar to Figure 8), or the value of individual system state variables. In the former case, the arcs indicate how transitions can be made between states; in the latter case, an arc indicates how state variable can influence another.

Numerous examples of digraph models for nuclear power plant system analysis can be found in the literature (e.g., [16,17]). In most such cases, the digraph is constructed during an intermediate step of the analysis. First, the system is modeled with a digraph. Second, fault trees are constructed from the digraph to determine how specific perturbations may arise.

The Logic Flowgraph Methodology (LFM) discussed in Ref. 16 appears to be one of the most promising of the digraph analysis methods, due to its simultaneous treatment of process variables and hardware states. However, because it is used to produce input to a fault tree, it requires the discretization of the process variables into a small number (5) of discrete, qualitative levels. The definition of these levels (typically {Very Low, Low, Nominal, High, Very High}), and the propagation of level disturbances through the system, is accomplished with significant amounts of judgment. Thus, LFM models are often difficult to generate and tend to be extremely difficult to review. It should be emphasized that the methodology is designed to assist an operator in on–line disturbance analysis (i.e., fault diagnosis), rather than to handle a full dynamic accident from beginning to end. Thus, it is tailored to deal with system perturbations; large changes over time are not considered.

Ref. 8 develops an alternative methodology, called the quantitative digraph (QD) approach, in an attempt to apply an LFM–like approach to accident sequence analysis. The objective of the model is to determine how the system can proceed through a given state space (e.g., {Pressure, Temperature, Mass}) to an undesired end state by hardware changes, and how long it takes to do this. Like LFM, the QD approach applies a varied set of operators for its nodes. Unlike LFM, the process variables are treated quantitatively (they can be discretized arbitrarily). System equations are used to determine possible transitions between discrete levels in the process variables (depending on which systems succeed or fail in the next time interval). This creates a data base linking transitions to hardware state changes, a data base which can be used when attempting to determine how a particular transition (or series of transitions) can arise.

The QD method is very similar to the DYLAM method, discussed later in this report. However, it is deductively oriented, whereas DYLAM is inductive. The problem with QD, therefore, is that it requires extremely large amounts of processing (transitions cannot be ignored *a priori* because their likelihood cannot be determined outside of the context of a scenario). Like static expanded event trees, therefore, digraphs appear to be impractical for general use in accident sequence analyses involving explicit treatment of process variables.

4.3.3 Markov Modeling

Markov models can be viewed as specialized digraph models, where the nodes represent the system states, and the arcs represent the transitions between states. Unlike the digraphs discussed above, time delays during transitions are modeled explicitly (they are assumed to be random variables with exponential distributions). The essential feature of Markov models is that the system is assumed to be memoryless; transitions are assumed only to depend on the current state (and not how the system arrived in the current state). Markov models have been widely applied to systems where repairs can be made; applications to large nuclear systems are discussed in Ref. 18.

An interesting application of Markov modeling to the problem of system analysis, including the treatment of process variables, has been developed by Aldemir [9,19,20]. In this approach, the system states are defined in terms of process variables and component configuration. The system dynamics are modeled using a discrete space–discrete time representation. The object of the majority of the analysis is to develop the matrix **M**:

$$\mathbf{M} \equiv \{m_{i,j}\} \quad 1 \leq i,j \leq N \tag{1}$$

where

$m_{i,j} \equiv P\{\text{transition from state i to state j in next } \Delta t \,|\, \text{in state i}\}$

$N \equiv$ number of states

The likelihood of being in any given state at time t can then be found using:

$$\underline{P}(t) = \mathbf{M} \cdot \underline{P}(t - \Delta t) \tag{2}$$

where $\underline{P}(t)$ is the vector of probabilities for being in each of the N states at time t.

The physical equations governing system behavior are incorporated when determining the $m_{i,j}$. Thus, this approach can handle component state changes that are deterministically triggerred by the process physics by including them in the system equations (e.g. PORV opening when setpoint pressure is reached) and those state changes caused by random transition independent of the process physics (e.g. failures and repairs). Ref. 9 presents an algorithm that makes the mechanization of matrix construction possible.

In general, the Markov approach can be summarized as follows.

a. Define events of interest. The events of interest are defined by specifying the allowed range of values that process variables may take (e.g., tank level must remain in a given range).

b. Identify the structural units (systems or components) and their possible states. This step evaluates component behaviors relative to the process variables. It may require the conduct of failure modes and effects analysis to help ensure completeness of the system study.

c. Prepare database or system model. When a system simulation model is already available, the analyst can use it to generate data needed for the probability calculation. When such model is not available, this step involves the creation of a system failure model. The model consists of algebraic or differential equations that describe dynamic hardware and process variable interactions and of the state space representation of the system behavior.

d. Develop transition matrix, using failure rate and repair rate data for structural units, and the system model (to provide constraints on possible transitions).

e. Evaluate probability vector $\underline{P}(t)$.

To simplify the application of this methodology to the storage tank problem described earlier, Ref. 9 makes the following assumptions.

1. The probability that the water level is anywhere within a given control range (e.g., $\alpha_1 \leq x \leq \alpha_2$) is constant. Thus, the actual value of the water level becomes irrelevant as long the interval is known.
2. Control units (i.e., the pumps and valve) do not change states in the interval $[t, t+\Delta t)$. Unit states may change instantaneously at $t+\Delta t$.

The step by step application to the example system of the Markov method is then as follows.

a. Define event of interest. The events of interest are dryout ($x < a$) and overflow ($x > b$).
b. Identify structural units and their possible states. There are three control units. Each unit has two states: operational and failed. The unit operational state as a function of water level is shown in Table 3.
c. Prepare system model. Based on the additional assumption #1 and the control rules (Table 3), it is convenient to divide the control region into three control intervals as shown in Table 4. Each control interval may be further divided into subintervals if the analyst deems it appropriate or necessary. Table 5 shows the state of each control unit in each interval. The water level change rate equations for the each component states combination are shown in Table 6. These relationships describe the system physics.
d. Evaluate transition matrix **M**. See Appendix C.
e. Evaluate probability. This simply applies Eq. (2).

The equations needed to construct the transition matrix are given in Appendix C. These equations have been implemented in a computer program and compared against the published results of Ref. 9 with reasonable success.

The primary strengths of the Markov approach described above are that: it explicitly accounts for process variable variation and control laws modifying system configuration according to process variable variations, and that it incorporates time explicitly. An additional characteristic is that, like other analytically based approaches, it calculates results for rare event sequences as easily as it does for likely event sequences. This can be contrasted with the results obtained from simulation approaches, where rare event sequences may not be sampled at all.

The primary weakness of the approach is that it requires explicit evaluation of the transition matrix **M**. This matrix can get unmanageably large for even a modest number of process variables and component states; if there are N process variables, each with n_i levels, and M components, each with m_j states, the matrix is of dimension

$$\left[\prod_{i=1}^{N} n_i \cdot \prod_{j=1}^{M} m_j\right] \text{ by } \left[\prod_{i=1}^{N} n_i \cdot \prod_{j=1}^{M} m_j\right]$$

It should also be noted that, if the system structure changes dynamically, the matrix must be reevaluated for each time step if the transition probabilities change dynamically.

The problem of large transition matrices in systems not involving process variables is generally treated by using state–merging and truncation techniques (e.g., [18,21,22]). These need to be explored in the context of systems involving continuous variables.

A second disadvantage of the approach is inherent in its assumptions; it neglects past events when computing transition probabilities. In the case of simple physical systems, this is often a reasonable approximation. However, it seems likely that in the course of an accident sequence, past history will be important because this will affect operator decisions. Thus, this assumption is a limitation when operator models are to be integrated into the analysis.

A third problem, less important than the first two, is that the procedure to mechanically generate the transition matrix can become extremely complicated when there are more than three process variables involved. This is largely a problem of dividing the control region into disjoint partitions and defining failure values of variables within the partition boundary.

4.3.4 Dynamic Logical Analytical Methodology (DYLAM)

The last methodology to be reviewed is like the event tree methodology, and different from the digraph and Markov methodologies, in that it is inductive, rather than deductive. Unlike the event tree methodology, it is also simulation oriented. In other words, the model is developed dynamically as it is executing according to a built–in set of rules. Accident scenarios are therefore defined implicitly (via the rules), rather than explicitly.

Dynamic Logical Analytical Method (DYLAM) is a simple simulation–based method for dynamic modeling and reliability assessment of a system [5,23–27]. In general, a system model is constructed by linking component models; these latter consist of equations for the process variables which are dependent on the component state. For example, a model for a pump may consist of an equation for the pump Δp as a function of the pump state. System failure is defined in terms of process variables (e.g., too little flow).

As applied to accident sequence analysis, the DYLAM approach can be summarized as follows. Starting at some inital state, the methodology requires the development, in principle, of all possible changes of state in a given Δt. The system equations are then used to update the values of process variables associated with each branch implied by a change in state. The process is repeated until all sequences being traced end at some absorbing state (e.g., core damage). Truncation rules are used to limit the size of the model as it is being constructed.

The main steps of the DYLAM analysis procedure are as follows:

1. Model all components in the system. This step identifies the component failure and degradation states, describes the analytical relationships involving the different process variables for all component states, and then assigns the component state transition rates or relationship.
2. Develop algorithm to resolve system equations. This step designs the method to solve the equations generated in the preceding step.
3. Define events of interest. This refers to the assignment of the system conditions which must be searched for. Top events can be defined in terms of values of process variables of interest, such as values which must not be exceeded to prevent undesired states. More than one top event can be studied in the same run.
4. Generate and analyze event sequences. This step generates possible event sequences using probabilistic or combinatorial rules. Starting with the initial conditions and sytem state, DYLAM updates the conditions, e.g. process variable values, and the likelihood of the state with time. When a time point or situation in which state change is likely is reached, DYLAM starts a new path incorporating the possible state change and appropriate conditions and, at the same time, maintains the initial path. Then, DYLAM continues to update the paths and acts similarly when either or both paths encounter point of possible state

change. DYLAM periodically compares the process variable values with the desired values and the current time with the maximum simulation time. Time points or situations of potential state change can be determine through probabilistic rules or conditioned on process variable values.

DYLAM can therefore be simply described as a simulation–based dynamic event tree generator. Event tree branchings occur at discrete points in time. Each branching can (and usually should) result in multiple branches, each branch corresponding to a possible set of system changes.

To avoid event sequence explosion, Ref. 5 establishes cut–off rules. Cut–off rules include preassigning a sequence probability level and prefixing the order of sequences to be generated. The order of the sequence refers to the number of failed states in a sequence. Branching rules, such as allowing branching out to occur only at time points when the probability of the sequence has dropped below a preset fraction of the initial sequence probability, also helps preclude event sequence explosion from occurring.

The step by step application of DYLAM to the example problem is described below.

1. Model all components in the system. Table 7 belows shows the results of the failure mode and effect analysis. "Operational" means that the control unit functions as prescribed by the control rules (Table 3). Table 8 gives the flowrate as a function of component state and water level. The system equations are as follows:

$$\dot{x}(t) = \dot{x}_1(t) + \dot{x}_2(t) + \dot{x}_3(t)$$

$$x(t + \Delta t) = x(t) + \dot{x}(t)\Delta t$$

2. Develop algorithm to resolve system equations. In this case, the equations are easily resolved without any special algorithm.
3. Define events of interest. The event of interest is the occurrence of either dryout ($x < a$) or overflow ($x > b$).
4. Generate and analyze event sequences. A simple code was written to simulate the process physics. The probability calculation, however, is not automated. In this case all control units have constant failure rates which are independent of the process physics (Table 7). DYLAM

handles a component with an independent constant failure rate by presetting a fraction of the initial reliability and allowing the component to branch at times when the reliability has dropped below the preset fraction of the initial reliability. After each branching, the initial reliability takes on a new value. Applying this rule to the example system, the three control units are assumed to branch out at the following times and their multiples:

Valve:	700 minutes
Pump 1:	810 minutes
Pump 2:	915 minutes

Table 9 shows the reliability of each component as a function of time (in minutes). Figure 10 shows an example on how to generate event sequences. Data on the event sequences that satisfy the events of interest appear in Table 10.

As in the case of the Markov modeling approach, DYLAM can handle dynamic aspects of the interaction between time dependent process variables, hardware, and control systems. Unlike the Markov analysis, DYLAM is general enough to allow modeling of scenarios in which past events can affect the likelihood of state transitions, and therefore can handle important aspects of operator behavior. DYLAM is also capable of dealing with arbitrary transition time distributions (the Markov analysis is limited to exponential transitions, unless matrix expanding techniques, e.g., the method of stages, are used).

DYLAM performs process simulation and reliability assessment in a self–contained way. This gives much flexibility in the use of DYLAM. Once the system model is installed, top events can be modified to analyze other events of interest.

Finally, DYLAM uses the physical equations governing system behavior to determine system success and failure. It eliminates the need of the analyst to judge success criteria (both number of components and duration of operation), especially in complicated cases (e.g., intermittent operation).

The primary problems with the DYLAM approach are those associated with simulation approaches. First, because truncation algorithms are used, it is not guaranteed to identify rare event sequences, even if these sequences have extremely large consequences. Second, because the model consists essentially of computer implementations of system equations, determining principal contributors to risk may be difficult, especially in cases where complicated control rules govern transitions. In other words, the model may appear to be too much like a "black box" to the user.

Of lesser importance, DYLAM calculations can be long and costly, since DYLAM explicitly solves the equations governing the process physics. Thus, an efficient algorithm used to resolve the system equations is highly desireable. Moreover, system models should be made simple without losing the essentials. The desire for simpler models, however, could affect the accuracy of the probability calculation.

In general, it appears that the problems associated with DYLAM are less fundamental than those associated with other methodologies. Truncation problems can be addressed, to an extent, simply by reducing the truncation limits imposed on a given run. The increase in computation time will be at least partially offset by the availability of increasingly fast and inexpensive computers. The issue of a "black box" model can be addressed by improving the sophistication of the model (e.g., adding tracers to key variables) and by developing sensitivity analysis procedures appropriate to the model. Our future work will therefore concentrate on developing DYLAM for a realistic application to nuclear power plant accident sequences, and on developing needed improvements to the DYLAM methodology.

5. REFERENCES

1. N.O. Siu and N.C. Rasmussen, "Physical Dependencies in Accident Sequence Analysis," Research Proposal for Grant NRC–04–88–143, Department of Nuclear Engineering, M.I.T., January 1988.

2. J.W. Minarick and C.A. Kukielka, "Precursors to Potential Severe Core Damage Accidents: 1969–1979, A Status Report," NUREG/CR–2497, ORNL/NOAC–182, December 1982.

3. J.W. Minarick, J.D. Harris, P.N. Austin, J.W. Cletcher, and E.W. Hagen, "Precursors to Potential Severe Core Damage Accidents: 1985, A Status Report," NUREG/CR–3591, ORNL/NOAC–232, December 1986.

4. A. Amendola, "Accident Sequence Dynamic Simulation Versus Event Trees," **Reliability Engineering and System Safety**, v. 22, 3–25(1988).

5. N.O. Siu and D.D. Lanning, "A Systems Model for Dynamic Human Error During Accident Sequences," Research Proposal for Grant NRC–04–89–356, Department of Nuclear Engineering, M.I.T., July 1989.

6. Nuclear Safety Analysis Center, Electric Power Research Institute, Duke Power Company, "Oconee PRA: A Probabilistic Risk Assessment of Oconee Unit 3," NSAC–60, June 1984.

7. Pickard, Lowe and Garrick, Inc., "Seabrook Station Probabilistic Safety Assessment," prepared for Public Service Company of New Hampshire and Yankee Atomic Electric Company, PLG–0300, December 1983.

8. S. Nguyen, "Incorporating Pysical Dependencies into Accident Sequence Analysis," S.M. Thesis, Department of Nuclear Engineering, M.I.T., May 1989.

9. T. Aldemir,"Computer–Assisted Markov Failure Modeling of Process Control Systems," **IEEE Transactions on Reliability**, v. R–36, 133–144(April 1987).

10. United States Nuclear Regulatory Commission,"PRA Procedures Guide," Review Draft, NUREG/CR–2300, 1981.

11. United States Nuclear Regulatory Commission, "Reactor Safety Study," WASH–1400 (NUREG–75/014), 1975.

12. C.J. Hsu, R. Youngblood, R. Fitzpatrick, and P. Amico, "Technical Evaluation Report: Assessment of the Risk Significance of 'Category C' Events in B&W Plants," Draft, August 1987.

13. G. Heslinga, "Human Reliability Analysis Using Event Trees," Kema Scientific & Technical Reports, v. 1, no. 3, 1983.

14. G. Apostolakis and T. L. Chu, " Time–Dependent Accident Sequences Including Human Actions," **Nuclear Technology**, v. 64, 115–126(February 1984).

15. United States Nuclear Regulatory Commission,"Reactor Risk Document," Draft, NUREG–1150, 1987.

16. S. Guarro and D. Okrent, "The Logic Flowgraph: A New Approach to Process Failure Modeling and Diagnosis for Disturbance Analysis Applications," **Nuclear Technology**, v. 67, 348–354(December 1984).

17. E.J. Henley and H. Kumamoto, **Reliability Engineering and Risk Assessment**, Prentice–Hall, N.J., 1981.

18. I. Papazoglou and E. Gyftopoulos,"Markov Processes for Reliability Analyses of Large Systems," **IEEE Transactions on Reliabilty**, v. R–26, 232–237(August 1977).

19. T. Aldemir,"Quantifying Setpoint Drift Effects in the Failure Analysis of Process Control Systems," **Reliability Engineering and System Safety**, v. 24, 33–50(January 1989).

20. T. Aldemir and M. Hassan,"A Data Base Oriented Dynamic Methodology for the Failure Analysis of Closed Loop Control Systems in Process Plants," Ohio State University, 1989.

21. M.I. Shooman and A.E. Laemmel,"Simplifications of Markov Models by State Merging," 1987 Proceedings of the Annual Reliabilty and Maintainability Symposium, Philadelphia, PA, 160–164, January 1987.

22. C. Singh,"Reliability Calculations of Large System," Proceedings of the 1975 Annual Reliability and Maintainability Symposium, Washington, D.C., 188–193, January 1975.

23. P.C. Cacciabue and A. Amendola,"Dynamic Logical Analytical Methodology Versus Fault Tree: The Case Study of the Auxiliary Feedwater System of a Nuclear Power Plant," **Nuclear Technology**, v. 74, 195–208, (August 1986).

24. Z. Nivolianitou, A. Amendola and G. Reina, "Reliability Analysis of Chemical Processes by the DYLAM Approach," **Reliability Engineering**, v. 14, 163–182(1986)

25. A. Amendola and G. Reina, "Event Sequences and Consequences Spectrum: A Methodology for Probabilistic Transient Analysis," **Nuclear Science and Engineering**, v. 77, 297–315(1981).

26. A. Amendola and G. Reina, "DYLAM–1: A Software Package for Event Sequences and Consequences Spectrum Methodology," EUR–9224 EN, CEC–JRC, 1984.

27. A. Amendola,"The DYLAM Approach to System Safety and Reliabilty Assessment," EUR–11361 EN, CEC–JRC, 1984.

TABLE 1. 1969-79 PRECURSORS REVIEWED FOR PHYSICAL DEPENDENCIES

PLANT NAME	DATE	PLANT TYPE	DESCRIPTION	DEPENDENCIES INVOLVED
Dresden 3	Dec 8, 1971	BWR	Safety valve open after loss of feedwater(LOFW)	Functional,Spatial
Dresden 2	May 5, 1970	BWR	Depressurization event	Functional,Spatial
Humbolt Bay	Jul 17, 1970	BWR	Loss of offsite power(LOOP)	Functional
Palisades	Sep 2, 1971	PWR	LOOP and failure of diesel generator to load	Functional
Lacrosse	Mar 24, 1971	BWR	Loss of power	Functional,Spatial
Nine Mile Point 1	Dec 31, 1971	BWR	High coolant level	Functional,Spatial,Human
Vermont Yankee	Dec 1, 1972	BWR	Loss of normal station power	Functional
Haddam Neck	Jan 19, 1974	PWR	LOOP	Functional
Browns Ferry 1	Mar 22, 1975	BWR	Cable tray fire caused extensive damage	Functional,Spatial
Pilgrim 1	Sep 13, 1975	BWR	LOOP and relief valve sticks open	Functional
Millstone 2	Jul 20, 1976	PWR	Apparent LOOP and failure of sfty related cmpnent	Functional
Millstone 1	Aug 10, 1976	BWR	Gas turbine fails during plant trip	Functional
Cooper	Aug 31, 1977	BWR	Loss of no-break-power and feedwater control	Functional
Cook 1	Sep 1, 1977	PWR	Reactor trip and LOOP	Functional
Davis-Besse 1	Nov 29, 1971	PWR	LOOP	Human
Calvert Cliffs 1	Apr 13, 1978	PWR	LOOP while shutdown	Functional
TMI-2	Apr 23, 1973	PWR	Multiple stuck-open relief valves	Functional
Rancho Seco	Mar 20, 1978	PWR	Failure of NNI and steam generator dryout	Functional
St. Lucie 1	May 14, 1978	PWR	LOOP during refuelling	Functional,Human
Beaver Valley 1	Jul 28, 1978	PWR	LOOP and diesel generator failure	Functional
Salem 1	Nov 27, 1978	PWR	Loss of vital instr bus-reactor trip	Functional
Oyster Creek	May 2, 1979	BWR	LOFW	Functional,Human
Hatch 1	Jun 3,1979	BWR	HPCI fails to start given LOFW	Functional,Human
St. Lucie 1	Sep 3, 1979	PWR	Switchyard lockout due to cable drop during storm	Functional
Davis-Besse 1	Oct 15, 1979	PWR	Reactor trip without LOOP	Functional
TMI-2 *	Mar 28, 1979	PWR	LOFW and open PORV	Functional,Human,Physical
Brunswick 1	Nov 20, 1979	BWR	RCIC turbine trip with HPCI unavailable	Functional

* indicates incident which involved physical dependency

TABLE 2. 1985 PRECURSORS REVIEWED FOR PHYSICAL DEPENDENCIES

PLANT NAME	DATE	PLANT TYPE	DESCRIPTION	DEPENDENCIES INVOLVED
Browns Ferry 1	Jan 16	BWR	LOFW and RCIC inoperability	Functional
Brunswick 1	Nov 2	BWR	Reactor islation&scram plus RCIC&DG trips	Functional
Calvert Cliffs 2	Apr 25	PWR	Stuck-opern atmospheric dump valve	Functional
Catawba 1*	Jun 22	PWR	LOFW&sec-side relief valve problems	Functional,Physical
Catawba 1	Jun 13	PWR	LOFW&sec-side relief valve problems	Functional
Davis-Besse 1*	Jun 9	PWR	LOFW & AFW failure	Functional,Human,Physical
Farley 2	Jul 15	PWR	Loss of power to 4160-V nonsafety buses	Functional
Grand Gulf 1	Dec 31	BWR	LOFW & HPCS failure	Functional
Hatch 1	Jan 6	BWR	Loss of HPCI/RCIC during recovery from a trip	Functional
Hatch 1*	May 15	BWR	Stuck-open relief valve&HPCI/RCIC unavailability	Functional,Physical
Hatch 2	Nov 5	BWR	LOFW & RCIC trip	Functional
La Salle 1	Feb 8	BWR	LOFW & RCIC trip	Functional
La Salle 1	May 31	BWR	Loss of circulating & nonsafety service water	Functional,Spatial
Mcguire 1	Jan 28	PWR	Reactor trip with stuck-open SG relief valve	Functional
Nine Mile Point	Nov 1	BWR	LOFW plus loss of one FWCI train & one ADS valve	Functional
Oconee 1	Apr 25	PWR	LOFW and stuck-open relief valve	Functional
Oyster Creek 1	Jun 12	BWR	MSIV closure&scram with subsequent SDV isoltion	Functional
Rancho Seco*	Dec 26	PWR	Loss of ICS and LOFW	Functional,Human,Physical
San Onofre 1	Jun 16	PWR	One train of HPI,AFW,&FWCI unavailable	Functional
San Onofre 1	Nov 21	PWR	Effective LOOP and AFW system unavailability	Functional,Human
Trojan	Mar 9	PWR	Plant trip,LOFW & water hammer	Functional
Trojan	Jul 20	PWR	AFW pumps fail on demand following trip	Functional
Turkey Point 3	Jul 22	PWR	Multiple AFW train failures following LOFW	Functional
Wolf Creek 1	Jun 9	PWR	Startup LOFW & AFW pump trip	Functional

* indicates incident which involved physical dependency

TABLE 3
OPERATIONAL UNIT STATES AS A FUNCTION OF WATER LEVEL

	Control Unit State		
Water Level	Valve	Pump1	Pump 2
$\alpha_1 \leq x \leq \alpha_2$	on	on	off
$\alpha_2 < x$	on	off	off
$x < \alpha_1$	off	on	on

TABLE 4
CONTROL INTERVALS

Interval Number (r)	Interval (V_r)
1	$a \leq x < \alpha_1$
2	$\alpha_1 \leq x \leq \alpha_2$
3	$\alpha_2 < x \leq b$

TABLE 5
OPERATIONAL AND FAILED UNIT STATES IN V_r

	Operational States			Failed States		
Interval	Valve	Pump1	Pump2	Valve	Pump1	Pump2
V_1	2	1	1	1	2	2
V_2	1	1	2	2	1	1
V_3	1	2	2	2	1	1

1 - open or on ; 2 - closed or off

TABLE 6
WATER LEVEL CHANGE RATE AS A FUNCTION OF CONTROL UNIT STATES AND CORRESPONDING UNIT STATE COMBINATION (n)

Control Unit State			n	Rate of Level		n_k	
Valve	Pump1	Pump2		Change	n_1	n_2	n_3
on	on	on	1	$-\dot{x}_1 + \dot{x}_2 + \dot{x}_3$	1	1	1
on	on	off	2	$-\dot{x}_1 + \dot{x}_2$	1	1	2
on	off	on	3	$-\dot{x}_1 + \dot{x}_3$	1	2	1
on	off	off	4	$-\dot{x}_1$	1	2	2
off	on	on	5	$\dot{x}_2 + \dot{x}_3$	2	1	1
off	on	off	6	$\dot{x}_2$	2	1	2
off	off	on	7	$\dot{x}_3$	2	2	1
off	off	off	8	0	2	2	2

n_k = 1 , open or on; 2, closed or off

TABLE 7
COMPONENT STATES AND TRANSITION RATES

Component	Condition/State	Probability
Valve	1 - operational	
	2 - failed	5.2 x 10-5/min
Pump1	1 - operational	
	2 - failed	7.6 x 10-5/min
Pump2	1 - operational	
	2 - failed	9.5 x 10-5/min

TABLE 8
COMPONENT FLOWRATES

Component	Water Level	State	Flowrate $\dot{x}_k(t)$
Valve	$\alpha_1 \leq x \leq \alpha_2$	1	$\dot{x}_1$
	$\alpha_1 \leq x \leq \alpha_2$	2	0
	$\alpha_2 < x$	1	$\dot{x}_1$
	$\alpha_2 < x$	2	0
	$x < \alpha_1$	1	0
	$x < \alpha_1$	2	$\dot{x}_1$
Pump1	$\alpha_1 \leq x \leq \alpha_2$	1	$\dot{x}_2$
	$\alpha_1 \leq x \leq \alpha_2$	2	0
	$\alpha_2 < x$	1	0
	$\alpha_2 < x$	2	$\dot{x}_2$
	$x < \alpha_1$	1	$\dot{x}_2$
	$x < \alpha_1$	2	0
Pump2	$\alpha_1 \leq x \leq \alpha_2$	1	0
	$\alpha_1 \leq x \leq \alpha_2$	2	$\dot{x}_3$
	$\alpha_2 < x$	1	0
	$\alpha_2 < x$	2	$\dot{x}_3$
	$x < \alpha_1$	1	$\dot{x}_3$
	$x < \alpha_1$	2	0

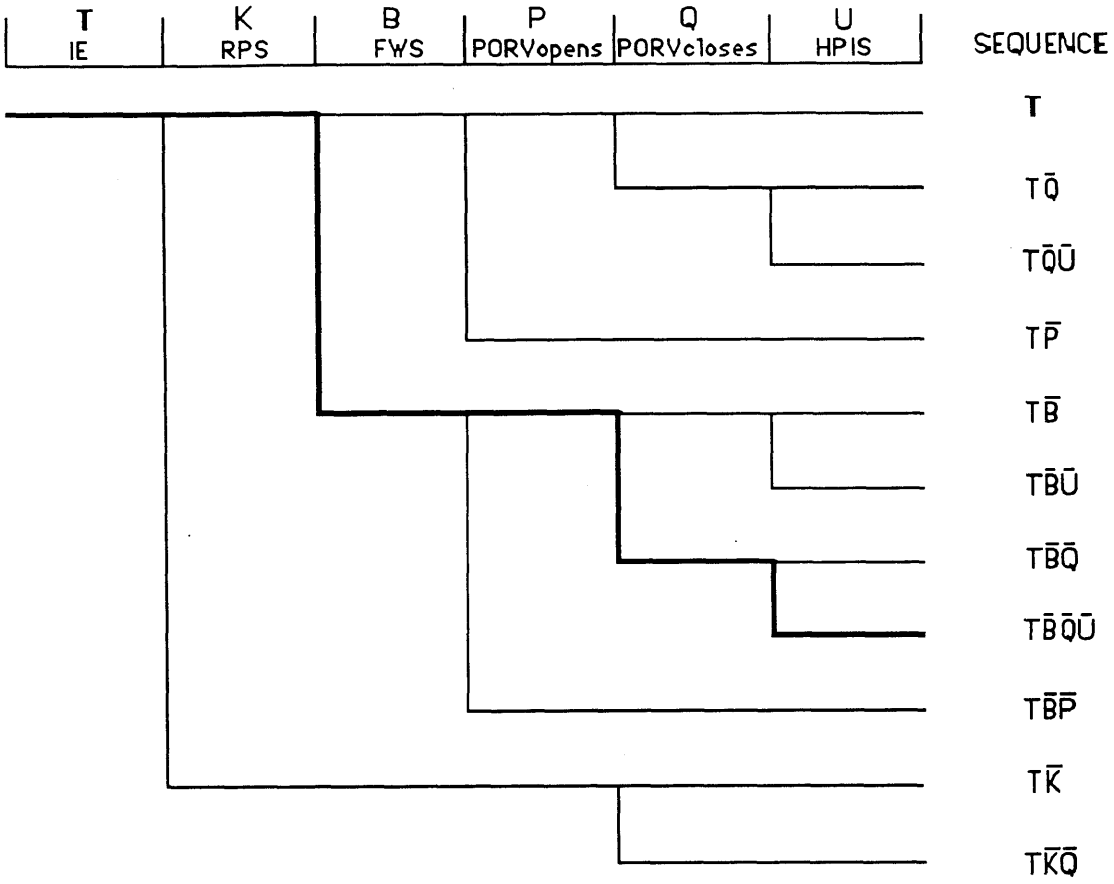
TABLE 9
RELIABILITY AS A FUNCTION OF TIME

Valve		Pump 1		Pump 2	
Time	Reliability	Time	Reliability	Time	Reliability
0-699	1	0-809	1	0-914	1
700-1399	.9645	810-1619	.94030	915-1829	.91675
1400-2099	.92979	1620-2429	.88416	1830-2744	.84042
2100-2799	.89655	2430-3239	.83137	2745-3658	.77045

TABLE 10
DATA ON SEQUENCES REACHING DRYOUT OR OVERFLOW

Sequence Number	System State	Event	Time Event Attained	Sequence Probability
1	V-1;P1-1;P2-1	Overflow	978	.000178
2	V-1;P1-1;P2-0	Overflow	1009	.001957
3	V-1;P1-0;P2-1	Overflow	1312	.002799
4	V-1;P1-1;P2-0	Overflow	1819	.003797
5	V-1;P1-0;P2-1	Overflow	2227	.005044
6	V-1;P1-1;P2-0	Overflow	2629	.005067
8	V-1;P1-1;P2-1	Dryout	1600	.0003489
9	V-1;P1-1;P2-1	Dryout	2300	.0005141
11	V-1;P1-1;P2-0	Dryout	1797	.003843
12	V-1;P1-1;P2-1	Dryout	2300	.0009856
16	V-1;P1-1;P2-1	Overflow	1681	.0006771
17	V-1;P1-0;P2-1	Overflow	1799	.005168
18	V-0;P1-1;P2-1	Overflow	2019	.008967
19	V-1;P1-1;P2-1	Overflow	2455	.001452
20	V-1;P1-0;P2-1	Overflow	2499	.007160
23	V-1;P1-1;P2-0	Overflow	1819	.007456
24	V-1;P1-0;P2-1	Overflow	2227	.009906
25	V-1;P1-1;P2-0	Overflow	2630	.009950
27	V-1;P1-1;P2-1	Dryout	2300	.001388
31	V-1;P1-1;P2-1	Overflow	2454	.002021
32	V-1;P1-0;P2-1	Overflow	2498	.009963
35	V-1;P1-1;P2-0	Overflow	2629	.01466

0 - operational ; 1 - failed



T - transient initiating event
K - reactor trip
B - heat removal via steam generator

P - PORV opens
Q - PORV recloses
U - high pressure injection system

Figure 1. TMI-2 Incident Using Oconee Event Tree

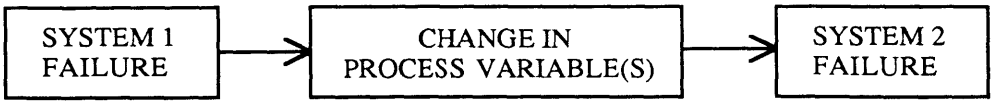


Figure 2. Direct Physical Dependency

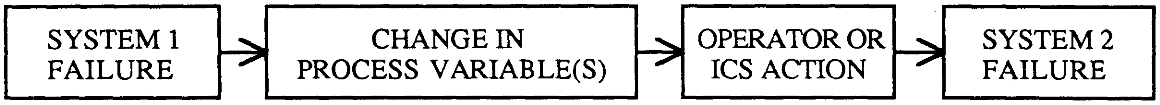


Figure 3. Indirect Physical Dependency

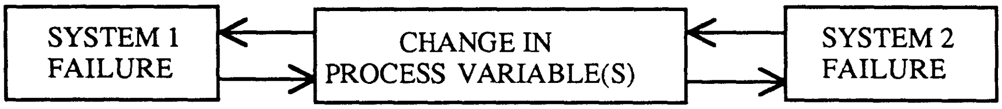
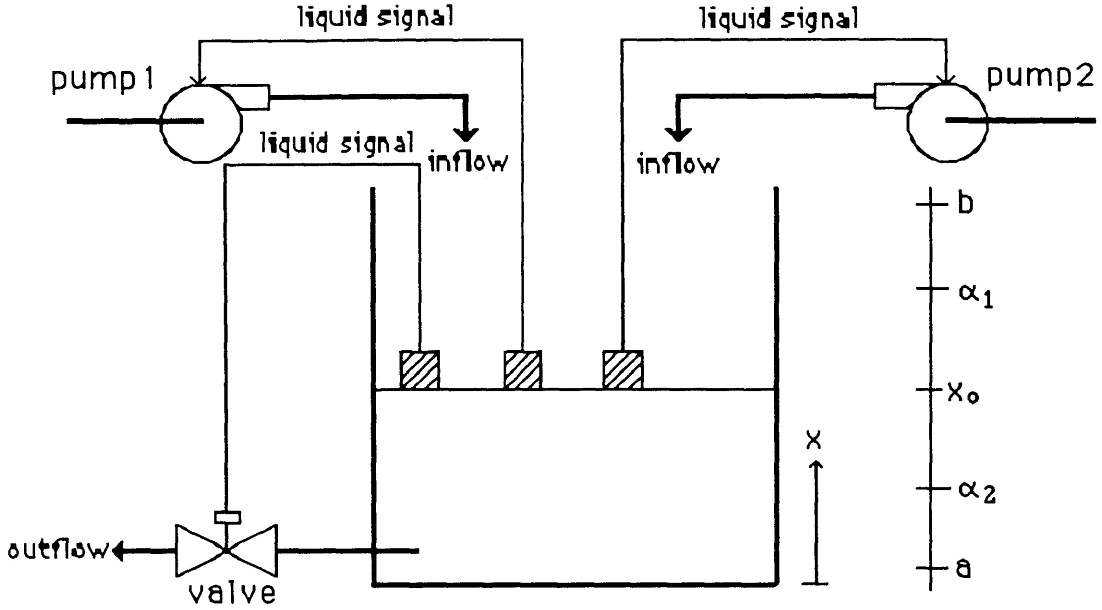


Figure 4. Cyclical Physical Dependency



x(t),x	- water level
$\dot{x}(t),\dot{x}$	- time rate of change of water level
a,b	- lower and upper bounds on water level respectively
α_1, α_2	- set points
x_o	- nominal water level , $\alpha_1 \leq x_o \leq \alpha_2$
t	- time

Figure 5. The Hypothetical Example System

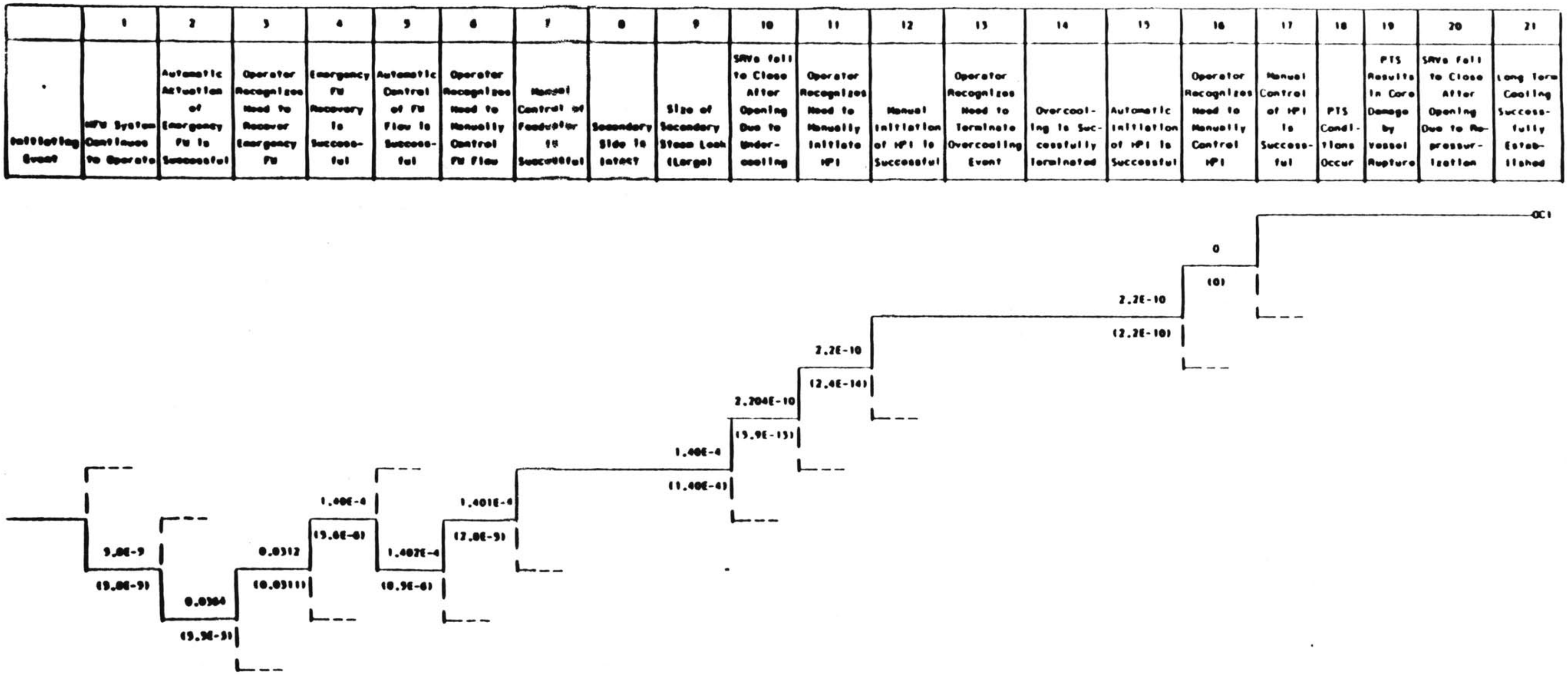

Initiating Event
1 MFW System Continues to Operate
2 Automatic Actuation of Emergency FW Is Successful
3 Operator Recognizes Need to Recover Emergency FW
4 Emergency FW Recovery Is Successful
5 Automatic Control of FW Flow Is Successful
6 Operator Recognizes Need to Manually Control FW Flow
7 Manual Control of Feedwater Is Successful
8 Secondary Side Is Intact
9 Size of Secondary Steam Leak (Large)
10 SRVs Fail to Close After Opening Due to Undercooling
11 Operator Recognizes Need to Manually Initiate HPI
12 Manual Initiation of HPI Is Successful
13 Operator Recognizes Need to Terminate Overcooling Event
14 Overcooling Is Successfully Terminated
15 Automatic Initiation of HPI Is Successful
16 Operator Recognizes Need to Manually Control HPI
17 Manual Control of HPI Is Successful
18 PTS Conditions Occur
19 PTS Results In Core Damage by Vessel Rupture
20 SRVs Fail to Close After Opening Due to Repressurization
21 Long Term Cooling Successfully Established
9.0E-9
(9.0E-9)
0.0304
(5.9E-3)
0.0312
(0.0311)
1.40E-4
(5.0E-8)
1.402E-4
(8.9E-8)
1.401E-4
(7.0E-9)
1.40E-4
(1.40E-4)
2.204E-10
(5.9E-13)
2.2E-10
(2.4E-14)
2.2E-10
(2.2E-10)
0
(0)
OC1

Figure 6. A B&W Davis-Besse LOFW Event Tree

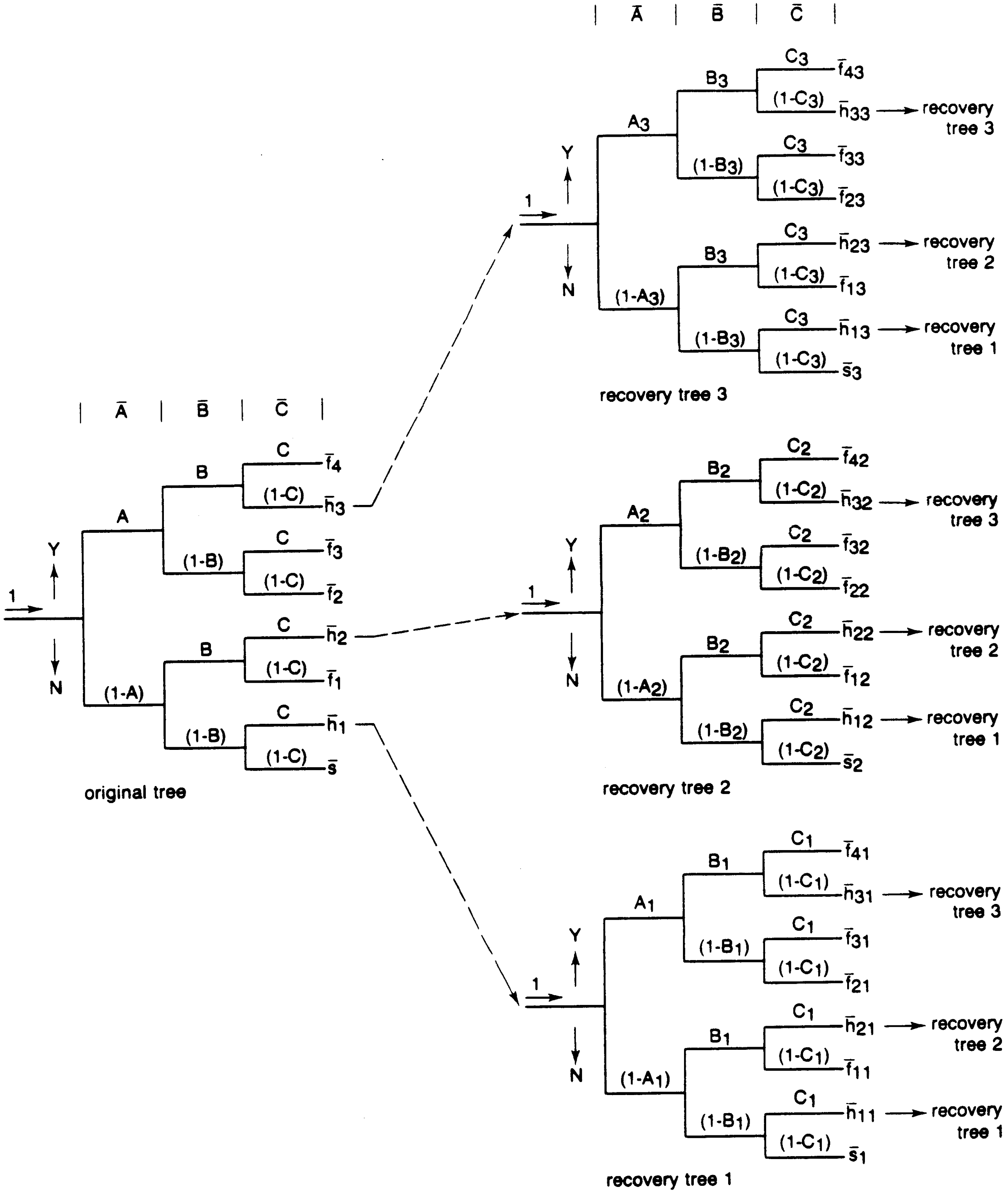


Figure 7. A Sample Use of Event Tree in Human Reliability Analysis

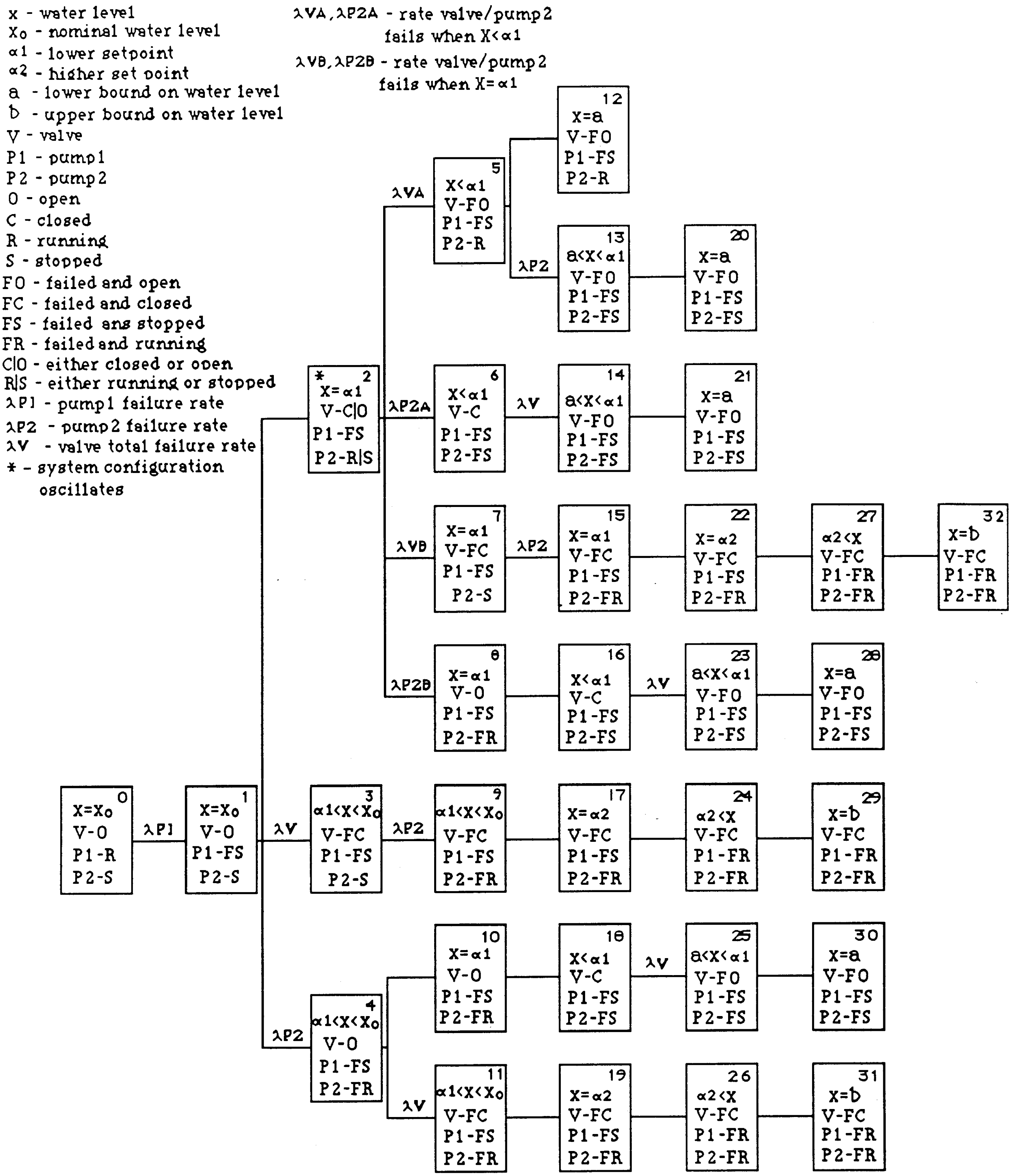


Figure 8. The State Transition Diagram of the System

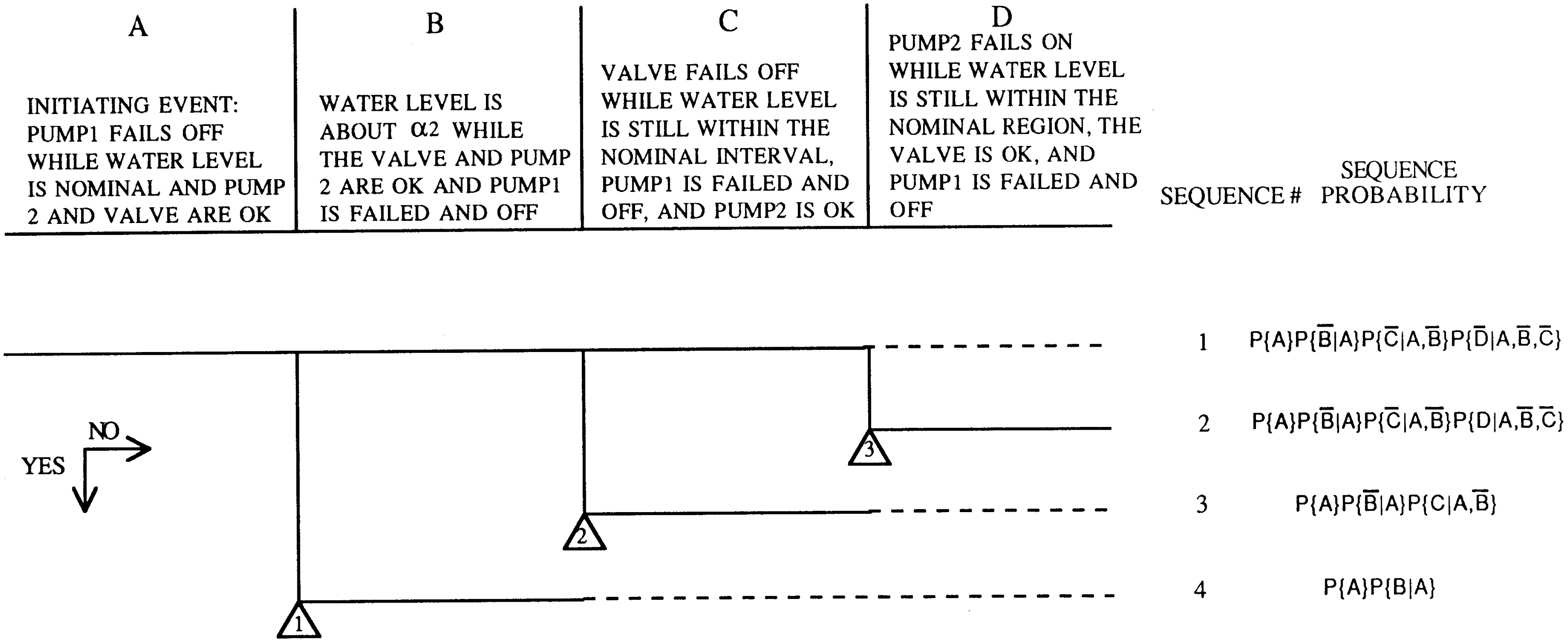
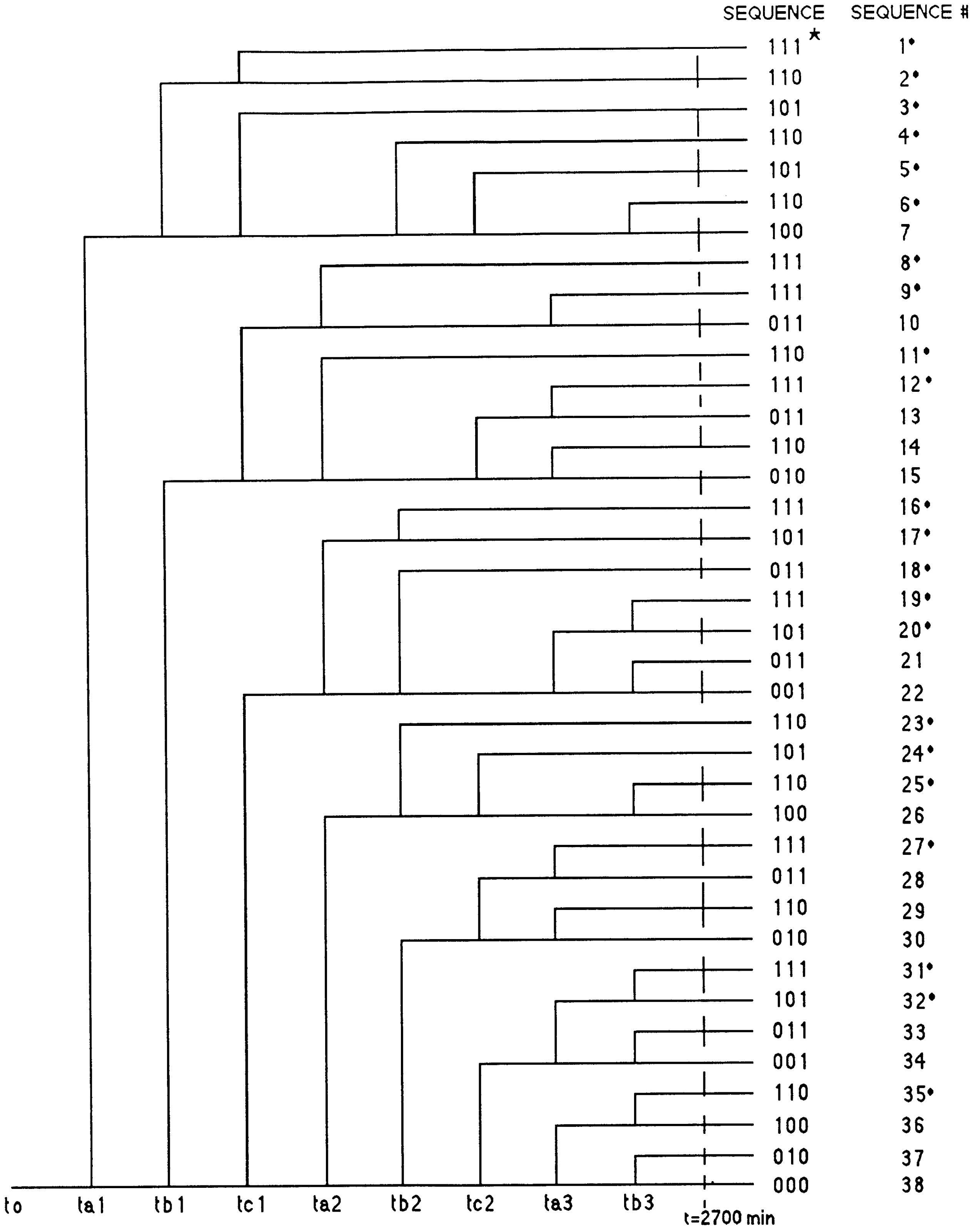

A
INITIATING EVENT: PUMP1 FAILS OFF WHILE WATER LEVEL IS NOMINAL AND PUMP 2 AND VALVE ARE OK
B
WATER LEVEL IS ABOUT $\alpha 2$ WHILE THE VALVE AND PUMP 2 ARE OK AND PUMP1 IS FAILED AND OFF
C
VALVE FAILS OFF WHILE WATER LEVEL IS STILL WITHIN THE NOMINAL INTERVAL, PUMP1 IS FAILED AND OFF, AND PUMP2 IS OK
D
PUMP2 FAILS ON WHILE WATER LEVEL IS STILL WITHIN THE NOMINAL REGION, THE VALVE IS OK, AND PUMP1 IS FAILED AND OFF
SEQUENCE #
SEQUENCE PROBABILITY
NO
YES
1
2
3
1 $P\{A\}P\{\overline{B}|A\}P\{\overline{C}|A,\overline{B}\}P\{\overline{D}|A,\overline{B},\overline{C}\}$
2 $P\{A\}P\{\overline{B}|A\}P\{\overline{C}|A,\overline{B}\}P\{D|A,\overline{B},\overline{C}\}$
3 $P\{A\}P\{\overline{B}|A\}P\{C|A,\overline{B}\}$
4 $P\{A\}P\{B|A\}$

Figure 9. Expanded Event Tree Covering Boxes 1 to 4



to = 0 min
ta1 = 700 min
tb1 = 810 min
tc1 = 915 min
ta2 = 1400 min

tb2 = 1620 min
tc2 = 1830 min
ta3 = 2100 min
tb3 = 2430 min

★the numbers indicate the states of the system; the first, second , and third digits correspond to the states of valve, pump 1, and pump 2 respectively (0 - good, 1 - failed)

• indicates that dryout or overflow is reached before t = 27000 min

Figure 10. DYLAM Event Sequence Generation

Appendix A. Questions which Served as Bases for the Headings of Surry Containment Event Tree

1. Is AC power available after the initiating event?
2. Does emergency core cooling fail prior to over pressurization of the containment?
3. What is the level of pre-existing containment leakage or isolation failure?
4. Where is the initial reactor coolant system break?
5. What is the size of the initial coolant system break?
6. Is the containment initially bypasssed?
7. Are the steam generators wet or dry?
8. Do the fan coolers fail to actuate before the core degradation?
9. Do the containment sprays fail to actuate in the injection mode before core degradation?
10. Do the containment sprays fail to actuate in the recirculation mode before core degradation?
11. To what degree, if any, is the auxiliary building initially bypassed?
12. Where, if at all, is there a temperature-induced failure of the reactor coolant system during the period of core degradation?
13. What is the size of the reactor coolant system failure?
14. What is the primary system pressure during the core degradation? Also, what would be the containment pressure increment from a primary system blowdown?
15. At what level, if any, is containment bypassed during core degradation?
16. What is the rate of blowdown to the containment during core degradation?
17. Do the containment sprays fail to actuate during the period of core degradation?
18. Is there containment heat removal during core degradation?
19. At what level, if any, does containment fail due to steam pressurization before vessel breach?
20. What is the containment pressure before vessel breach?
21. Is there a hydrogen burn before vessel breach? Also, what is the pressure increment from such a burn?
22. Does containment fail because of a hydrogen burn before vessel breach? Also, what is the associated failure pressure and standard deviation?

23. To what degree, if any, is the auxiliary building bypassed before vessel breach?
24. Do the fan coolers fail after the early hydrogen burn?
25. Do the containment sprays fail after the early hydrogen burn?
26. Is there containment heat removal after the early hydrogen burn?
27. What is the mode of vessel breach?
28. Does direct heating of the containment atmosphere occur just after vessel breach? Also, what is the pressure increment from a steam spike alone and from a steam spike plus a direct heating?
29. Is there a hydrogen burn just after vessel breach? Also, what is the pressure increment from such a burn?
30. Does containment fail due to a steam pike, direct heating, and/or hydrogen burn just after vessel breach?
31. What is the mode of containment failure, if any, just after vessel breach?
32. To what degree, if any, is the auxiliary building bypassed just after vessel breach?
33. Do the fan coolers fail after vessel breach?
34. Do the containment sprays fail after vessel breach?
35. Is there an oxidation release?
36. Is AC power restored after vessel breach?
37. Do the fan coolers fail late in the accident?
38. Do the containment sprays fail late in the accident?
39. Is there containment heat removal late in the accident?
40. Has the inventory of refuelling water storage tank been injected into containment?
41. What is the amount of water injected into containment?
42. Do significant core-concrete interactions occur after vessel breach?
43. Is AC power recovered late in the accident?
44. What is the containment pressure late in the accident?
45. Has a large leak or gross failure occurred late in the accident?
46. Does a later hydrogen burn occur?
47. What pressure rise would occur if combustible gases were to burn late in the accident?
48. Would containment fail due to a late hydrogen burn or steam production?
49. In what way, if at all, does containment fail due to gradual pressurization from water boiloff or noncondensible gases or hydrogen burning (leakage or gross rupture)?

50. To what degree,if any, is the auxiliary building bypassed late in the accident?
51. Do the fan coolers fail after the late hydrogen burn?
52. Do the containment sprays fail after the late hydrogen burn?
53. Is there containment heat removal very late in the accident?
54. Does basemat meltthrough occur, given no prior containment failure?
55. Does containment depressurize after basemat meltthrough?
56. What is the ultimate containment failure mode, if any, resulting from core-concrete interactions?
57. To what degree, if any, is the auxiliary building bypassed very late in the accident?

Appendix B. Split Fraction Calculation for the Expanded Event Tree Covering Boxes 1 to 4

$t_\alpha = \dfrac{x_o - \alpha_1}{\dot{x}_v}$ - time it takes for water level to reach $\alpha 1$ given box 1 condition

t_v - time it takes for valve to fail given box 1 condition; t_v is exponentially distribyted

t_p - time it takes for pump 2 to fail given box 1 condition; t_p is exponentially distributed

P{1} - probability of branch 1 in Figure 9

P{2} - probability of branch 2 in Figure 9

P{3} - probability of branch 3 in Figure 9

$f_v(t_v), F_v(t_v)$ - probability density function and cumulative distribution function of t_v respectively

$f_p(t_p), F_p(t_p)$ - probability density function and cumulative distribution function of t_p respectively

$f(t) = (\lambda t)\exp(-\lambda t)$ - probability density function of an exponentially distributed random variable t

$F(t) = \int_0^t f(t)dt$ - cumulative distribution function of random variable t

$$
\begin{aligned}
P\{1\} &= P\{\ B \mid A\ \} \\
&= P\{\ t_v > t_\alpha, t_p > t_\alpha\ \} \\
&= [1 - F_v(t_v)]\ [1 - F_p(t_p)],\ t_v \text{ and } t_p \text{ are independent}
\end{aligned}
$$

$$
\begin{aligned}
P\{2\} &= P\{\ C \mid A, \overline{B}\ \} \\
&= P\{\ t_p > t_v \mid \min(t_v, t_p) < t_\alpha\ \} \\
&= P\{\ t > 0 \mid \min(t_v, t_p) < t_\alpha\ \},\ t = t_p - t_v \\
&= 1 - F_t(0)
\end{aligned}
$$

where,

$$
F_t(t) = \int_0^{t_\alpha} f_v(t_v)\ \Big[\ \int_0^{t+t_v} f_p(t_p)dt_p\ \Big]\ dt_v
$$

$$
\begin{aligned}
P\{3\} &= P\{\ D \mid A, \overline{B}, \overline{C}\ \} \\
&= P\{\ t_p < t_v \mid \min(t_v, t_p) < t_\alpha,\ t_p < t_v\ \} \\
&= 1.0
\end{aligned}
$$

Appendix C. Aldemir-Derived Probabilistic Equations

$$p_{n,r}(t+\Delta t) = \sum_{n'=1}^{N} \sum_{r'=1}^{5} q_{n,r}^{n',r'}(\Delta t)\, p_{n',r'}(t)$$

where

$$q_{n,r}^{n',r'}(\Delta t) = \left(\frac{1}{x_{r'}}\right) \prod_{k=1}^{3} c_k(n_k | n'_k, r' \to r, \Delta t) \int_{x_{r'}} dx'\ e_r(x' + \dot{x}\Delta t)$$

if $r'=1,2,3$; $r=1,2,2$

$$= \left(\frac{\delta_{n',n}}{x_{r'}}\right) \int_{x_{r'}} dx'\ e_r(x' + \dot{x}\Delta t)$$

if $r'=1,2,3$; $r=4,5$

$$= \delta_{n',n}\delta_{r',r}\ ,\ \text{otherwise}$$

$c_k(n_k | n'_k, r' \to r, \Delta t)$ = $1-\lambda_k\Delta t$, if unit is operational at t and at $t+\Delta t$
λ_k, if unit is operational at t but not at $t+\Delta t$
1, if unit is failed at t and at $t+\Delta t$
0, otherwise

r - subscript to indicate particular interval r, where $r = 1,2,3,4$(dryout), 5(overflow)

V_r - control interval r, $r = 1,2,3,4,5$

x_r - length of interval V_r

N - number of possible combinations of unit states, $N=8$

$n(t),n$ - designation number of control unit state combinations $n= 1,..,N$

n_k - control unit state, $k=1$(operational), 2(failed)

$p_{n,r}(t)$ - $\Pr\{n(t) = n,\ x(t) \in V_r\}$

$p(n,x,t)$ - $\Pr\{n(t) = n,\ x(t)=x\}$

$\delta_{n',n}$ - Kronecker delta, $\delta_{n',n} = 1$ if $n'= n$, 0 otherwise

$e_r(x)$ - step function, $e_r(x) = 1$ if $x \in V_r$, 0 otherwise

$q_{n,r}^{n',r'}(\Delta t)$ - Markov transition matrix elements

- $\Pr\{x(t+\Delta t)\in V_r | x(t)\in V_{r'}, n(t)=n'\}$
$\Pr\{n(t+\Delta t) = n |\ n(t)=n', x(t+\Delta t)\in V_r, x(t)\in V_{r'}\}$

$c_k(n_k | n_{k'}, r' \to r, \Delta t)$ - $\Pr\{n(t+\Delta t) = n_k |\ n(t)=n_{k'}, x(t)\in V_{r'} \to x(t+\Delta t)\in V_r\}$