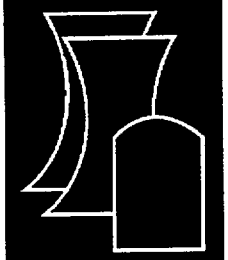
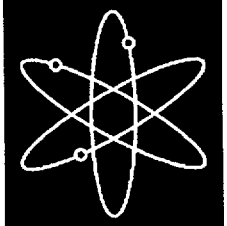
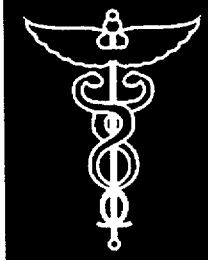
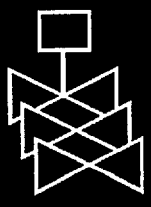
NUREG/CR-6710









Extending the Dynamic Flowgraph Methodology (DFM) to Model Human Performance and Team Effects

ASCA, Inc.

U.S. Nuclear Regulatory Commission
Office of Nuclear Regulatory Research
Washington, DC 20555-0001

AVAILABILITY OF REFERENCE MATERIALS IN NRC PUBLICATIONS

NRC Reference Material

As of November 1999, you may electronically access NUREG-series publications and other NRC records at NRC's Public Electronic Reading Room at www.nrc.gov/NRC/ADAMS/index.html.
Publicly released records include, to name a few, NUREG-series publications; *Federal Register* notices; applicant, licensee, and vendor documents and correspondence; NRC correspondence and internal memoranda; bulletins and information notices; inspection and investigative reports; licensee event reports; and Commission papers and their attachments.

NRC publications in the NUREG series, NRC regulations, and *Title 10, Energy*, in the Code of *Federal Regulations* may also be purchased from one of these two sources.

1. The Superintendent of Documents
   U.S. Government Printing Office
   Mail Stop SSOP
   Washington, DC 20402–0001
   Internet: bookstore.gpo.gov
   Telephone: 202-512-1800
   Fax: 202-512-2250
2. The National Technical Information Service
   Springfield, VA 22161–0002
   www.ntis.gov
   1–800–553–6847 or, locally, 703–605–6000

A single copy of each NRC draft report for comment is available free, to the extent of supply, upon written request as follows:
Address: Office of the Chief Information Officer,
Reproduction and Distribution
Services Section
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001
E-mail: DISTRIBUTION@nrc.gov
Facsimile: 301–415–2289

Some publications in the NUREG series that are posted at NRC's Web site address www.nrc.gov/NRC/NUREGS/indexnum.html are updated periodically and may differ from the last printed version. Although references to material found on a Web site bear the date the material was accessed, the material available on the date cited may subsequently be removed from the site.

Non-NRC Reference Material

Documents available from public and special technical libraries include all open literature items, such as books, journal articles, and transactions, *Federal Register* notices, Federal and State legislation, and congressional reports. Such documents as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings may be purchased from their sponsoring organization.

Copies of industry codes and standards used in a substantive manner in the NRC regulatory process are maintained at—

The NRC Technical Library
Two White Flint North
11545 Rockville Pike
Rockville, MD 20852–2738

These standards are available in the library for reference use by the public. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from—

American National Standards Institute
11 West 42nd Street
New York, NY 10036–8002
www.ansi.org
212–642–4900

Legally binding regulatory requirements are stated only in laws; NRC regulations; licenses, including technical specifications; or orders, not in NUREG-series publications. The views expressed in contractor-prepared publications in this series are not necessarily those of the NRC.

The NUREG series comprises (1) technical and administrative reports and books prepared by the staff (NUREG–XXXX) or agency contractors (NUREG/CR–XXXX), (2) proceedings of conferences (NUREG/CP–XXXX), (3) reports resulting from international agreements (NUREG/IA–XXXX), (4) brochures (NUREG/BR–XXXX), and (5) compilations of legal decisions and orders of the Commission and Atomic and Safety Licensing Boards and of Directors' decisions under Section 2.206 of NRC's regulations (NUREG–0750).

NUREG/CR-6710

Extending the Dynamic Flowgraph Methodology (DFM) to Model Human Performance and Team Effects

Manuscript Completed: January 2001
Date Published: March 2001

Prepared by
A. Milici, R. Mulvihill, S. Guarro

ASCA, Inc.
704 Silver Spur Road
Rolling Hills, CA 90274

J. J. Persensky, NRC Project Manager

Prepared for
Division of Systems Analysis and Regulatory Effectiveness
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001
NRC Job Code W6719

ABSTRACT

This report addresses the development of a structure for the modeling and analysis of control room teams to represent team related human errors of commission and omission in nuclear power plant accident scenarios. The structure includes the identification of unsafe actions (UAs) and error forcing contexts (EFCs) during abnormal or accident situations that can lead to a human failure event. This report also describes guidelines for the screening of sequences for which a dynamic flowgraph methodology (DFM) analysis could be effectively applied. The screening, DFM modeling and DFM analysis processes are demonstrated, via case studies, within the general human reliability analysis approach provided by the ATHEANA framework. In addition, this report describes extensions to the DFM methodology devised specifically to facilitate its application to teams effects modeling and analysis; in particular, it describes a library of pre-built DFM model modules that represent typical team-related cognitive, assessment and interaction processes.

Contents

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

This report presents a structure that can be used for modeling accident scenarios that contain errors of commission or omission as initiating events or intermediate events. These team related human error events are correlated with unsafe actions and associated error forcing contexts in a nuclear power plant team-oriented setting. The structure utilizes the Dynamic Flowgraph Methodology (DFM) and is meant for application within a general human reliability assessment approach, for example such as ATHEANA (Cooper, et.al., 1996; Parry, et.al, 1996). Within the context of nuclear power plant Probabilistic Risk Assessment (PRA), a human failure event (HFE) represents the failure of a plant function, system, and/or component which occurs as a result of an unsafe action or sequence of actions by plant operators or personnel, resulting in a degraded plant condition. A human failure event can be either an error of commission (EOC), in which the operators intentionally disable or terminate a necessary safety function or intentionally initiate an inappropriate system, or an error of omission, in which the operators fail to initiate a required safety system or function. In a PRA, in order to quantify an HFE it is necessary to identify the potential underlying unsafe actions and their associated error forcing contexts. Thus, the structure presented here consists of a step-by-step process of modeling and analysis that, given the definition of a human failure event, permits the systematic identification of the unsafe actions and error forcing contexts (EFCs) that may lead to the HFE. An unsafe action represents an action, or sequence of actions, inappropriately taken, or not taken when needed, by plant personnel that result in an HFE. An error-forcing context (EFC) represents the combined effect of performance shaping factors (PSFs) and plant conditions that create a situation in which an unsafe act becomes likely (or possible). PSFs concerned with team-related processes, such as communication and coordination, are explicitly considered as possible contributors to error-forcing contexts. Unsafe actions and their one or more associated PSFs can be included in the DFM model that represents the accident scenario. The structure consists of four basic, separate processes: a screening process, an information gathering process, a modeling process and an analysis process.

The screening process begins by examining the dominant accident sequences that are identified in a PRA followed by less dominant sequences that have the potential to be risk significant if team human errors can result in an additional initiating event(s) or intermediate (progression) event(s) related to safety system functions. One must also consider totally new accident sequences not yet represented in a PRA, particularly those that may result from errors of commission that result in an accident initiating event, followed by a progression of events that bears little similarity to events that are represented in the PRA.

In order to ensure coverage in a PRA of the types of safety related system functions that the screening process should apply to, a Team Human Error/Unsafe Action Master Logic Diagram (MLD) has been developed. Each of the functions in the MLD can be examined in accordance with generic event sequence diagrams to develop a generic accident event sequence. To convert the generic sequence to a power reactor specific sequence it is necessary to review the specific procedures, such as appropriate instrument readings, procedural steps, and particulars of potentially unsafe and safe shutdown conditions.

The result of the screening process is a set of initiating events that should be compared with the set of initiating events covered in the existing plant PRA to uncover any potentially significant HFE-initiated sequences that may have been omitted from the PRA.

The screening guidelines recommend the execution of the following screening process steps:

-- Review the Team Human Error/Unsafe Action MLD for applicability and make modifications as appropriate for a specific plant.

-- For each MLD safety related function, cross-reference to the generic ESDs and develop generic accident sequences.

-- Convert generic accident sequences to plant-specific accident sequences after a review of appropriate system drawings and specifications as well as team procedures.

-- Compare with dominant accident sequences identified in the PRA.

The information gathering process can be summarized as follows:

-- Identify safety system associated with HFE.

-- Identify the procedures that pertain to the accident sequence in question and the step(s) in the procedures relevant to the HFE.

-- Identify what information is required by the operators in order to follow the procedures and monitor the effectiveness of their actions.

-- Identify instruments that provide the information, either directly or indirectly.

-- Identify how the instruments can provide misleading information.

With respect to the modeling process, the DFM model consists of four main parts: the model of the plant, the model of the instruments, the model of the cognitive behavior of each member of the team and the model of the interaction between the team members.

The analysis of the model consists of three stages and two types of analysis. The two types of analysis are forward simulation and back-tracing to find prime implicants. The first stage of the analysis consists of using forward simulation to verify that the model makes sense. The second stage consists of using back-tracing, with the HFE as the top event, to identify prime implicants. The resulting prime implicants contain the unsafe actions and the associated error-forcing contexts. The third stage consists of using forward simulation with the prime implicants of interest as boundary conditions to understand how the unsafe actions and error forcing contexts lead to the HFE.

An important aspect of the structure presented here is the dynamic modeling of the cognitive activities associated with the four stages of information processing, monitoring and detection, situation assessment, response planning and response implementation.

Situation assessment is the activity of constructing a mental model based on observations. The operators' mental model consists of their understanding of the current plant state and behavior, which relates to their expectations of future plant states and behaviors as the accident progresses. The operators update their mental model based upon information received about the plant state, both in terms of system states and process conditions. This information is generally received, either directly or indirectly, through instrument readings or indications, resulting from changes in the plant state or process behavior due to the progression of the accident or due to operator actions. The operators' situation assessment then guides their development of future response plans and further monitoring activities. Monitoring and detection activities are either directed by procedures and the operators' mental model of the situation, or by alarms or other signals that get the operator's attention. Response planning is the process of deciding what actions to take. The operators' mental model of the plant state and the available procedures are used to formulate a response plan. Response implementation refers to the actual physical implementation of the response plan.

Two case studies, taken from actual nuclear power plant accident scenarios, were developed and analyzed to finalize, validate and demonstrate the DFM modeling and analysis approach. The development case study was conducted mainly to finalize the overall approach and to aid in the identification of common situations for which pre-built DFM model modules could be developed. The test case study was used to demonstrate and validate the entire DFM modeling structure for team related human error analysis.

Practical computational considerations to permit the application of the DFM modeling structure to complex human error and system interactions that involve a relatively large number of states include:

-- Increased computational efficiency within the 32 MB RAM solution environment.

-- Expansion of the computational efficiency and capability at the DFM algorithmic level by prioritizing and minimization of RAM usage.

-- Expansion of computer capability by writing the DFM code to utilize the Hard Drive (HD) memory.

It is believed that the use of such techniques will prevent any limitations on the complexity of team related accident scenarios that can be evaluated, within the structure presented in this document.

1. INTRODUCTION

Both industrial experience and probabilistic risk assessments (PRAs) have shown that human errors determine, to a large extent, the level of risk associated with operating nuclear power plants (NPPs) and other complex engineering systems. While most current risk analyses have focused upon a single operator acting alone, the fact is that most NPP-related decisions are made by teams of operators. The members of the team not only share information from different plant hardware, software, and human sources, but also use their training and knowledge base to contribute to the decision-making process through various forms of communication. Moreover, these group decisions are often made under time constraints and psychological pressures. The studies performed by behavioral scientists on team effects and the related subjects have focused on characterizing team effects as issues isolated from hardware and software systems, while in safety risk assessments the system failures caused by hardware, software, and human error have been addressed in an integrated, but limited, manner. The fact that the current risk assessments do not properly model the impacts of team effects on plant safety could cause a potential concern to the credibility of the results of such analyses.

Team members interact with the hardware of the system by receiving information from sensors and by intervening to control the physical processes that the system implements. Empirical and experimental studies, however, have provided evidence that internal team processes are important to the successful performance of the team.

The objective of the research reported here was to develop a framework for the use of the Dynamic Flowgraph Methodology (DFM) to model and analyze human performance in a control room team setting. Specifically, the research was concerned with the development of a framework for the identification of scenarios in which team behavior may play an important role, and given such a scenario, the identification of *unsafe actions* and *error forcing contexts* that, when occurring in combination, result in a *human failure event.* Another objective of the research was to identify and develop extensions to the existing DFM Software Toolset that would be useful for implementing the DFM modeling and analysis framework.

The current version of DFM is an extension of its predecessor, the Logic Flowgraph Methodology (LFM) (Guarro and Okrent, 1984), which models systems in steady-state. DFM, however, is dynamic in nature, and can thus capture the temporal relationships between system variables. The development of dynamic, multi-valued (non-binary) fault trees in DFM is a significant technical advance in system safety and reliability analysis (Garret, et. al. 1995a; Garret, et. al., 1995b; Guarro, et al, 1996; Yau, 1997). With the additional modules for human performance and team effects developed in this project, DFM can be used to diagnose and to reduce system faults resulting from combinations of human errors, software logic errors, hardware failures, and environmental conditions.

The DFM Software Toolset is a software environment developed for the representation and analysis of the cause-effect and timed relationships of complex systems. The graphic representation and logic analyzing capabilities of the DFM environment makes it a promising tool for modeling human performance and team-effects of operators working in complex system environments.

2. PROJECT OVERVIEW

The ultimate goal of this research was the extension of the DFM methodology, which has already been developed to conduct reliability and safety assessments of systems with complex software and hardware elements, to include human performance and team processes. The effort included the development of application procedures and guidelines, as well as a self-contained software package embodying these procedures and the functionality needed to realize the use of the approach. It is anticipated that the approach and associated software will be useful as a means of assuring the reliability and safety of a complex engineering system with respect to the interaction of hardware, software, and humanware.

The methodology developed in this project will enable one to identify some of the potential failure modes of the decision-making processes that are inherent to the operation of a process plant (e.g., NPP) under normal and accident situations and will include consideration of errors of commission and omission in full-power, low-power and shutdown cases. Team effects and dynamic features were among the major issues addressed in detail in the project by utilizing the DFM technique to model the problem. Although intended as basic research, the results of the project should be relevant to the NRC mission of improving the reliable and safe operation of NPPs. Furthermore, since the theme is a general one of probing into the cognitive processes involved in a team task under time constraints and psychological pressures, the use of the insights gained from this project are not limited to the nuclear power industry; the project results are expected to contribute to the general understanding of operator behavior in different settings, e.g., various sorts of control room conditions.

We note that DFM is modular in nature and, thus, not tied to a specific model of human behavior. To demonstrate its usefulness we have used it to implement certain specific models for cognitive behavior; however, this was for demonstration purposes only and other types of cognitive and human reliability models could have alternatively been used as well.

Deficiencies in PRAs can often be traced to the fact that, generally speaking, team errors of commission are not identified as separate initiating events for full power operation. Team errors of commission are thought to be inherent in the failure frequency assigned to the related hardware initiating event, such as reactor trip. Errors of commission have been incorporated into some shutdown PRAs. Several differences between shutdown and full-power cases derive form the fact that operator responses are often not as clearly guided by procedure in the former as in the full-power case. Thus, misdiagnosis is potentially of more concern for the shutdown case. In addition, the plant configuration is constantly changing during an outage, and there are many different activities proceeding in parallel, both inside and outside the control room. In the full-power case, the activities of concern are the control room crew's responses to initiating events. Faulty responses are either errors of commission or omission that relate to faulty recovery operations or misoperation of safety systems.

The overall approach is expected to allow significant contributions/improvements to be made in the areas of human reliability analysis, operator (individual and team) training, procedure writing/effectiveness, risk management, and cost effectiveness (i.e., through improvements in plant availability). For example, in the area of operator training, the fact that operators have to act as a team needs to be explicitly addressed. Furthermore, the decision tables and fault trees produced via DFM can be used not only to enhance the performance of root-cause analyses and the development of risk management strategies, but also to improve plant availability. Finally, a model of operator behavior that takes into account the dynamic team effects will be of great value to the design of computerized decision support systems, which are expected to be used to a great extent as the industry moves into new reactor designs, e.g., the advanced light water reactors (ALWRs).

2.1 Phase II Technical Objectives

This section discusses the objectives pursued in the Phase II research. The broad technical objective of this research was to extend the concepts and software of DFM and to develop and demonstrate the use of DFM as an integrated methodological approach with an associated set of software tools that can model, within one integrated environment, the hardware, software, and humanware elements of a complex system. Such an extended DFM environment can then be used to diagnose and to reduce system faults resulting from combinations of human errors, software logic errors, hardware failures, and environmental conditions.

2.1.1 Phase II Technical Objective 1: Finalization of the Methodology Developed in 2.1.1 Phase I of the Project

One technical objective of the Phase II research was to finalize the DFM modeling technique for human performance and team effects in complex systems. In the Phase I research the authors developed some concepts of how team human performance and team effects might be modeled using the DFM modeling approach and software. Included were such elements as mental models of system states and system behavior, decision making, action execution, communication and coordination. To solidify the approach developed in Phase I, formal guidelines for the DFM representation of the above elements were formulated, and consistent guidelines for how those elements relate to each other were developed.

2.1.2 Phase II Technical Objective 2: Development of Design Specifications for Extensions of DFM Software

A second technical objective of the Phase II research was to develop complete specifications for the detailed design of the functionality of the extensions to the DFM software. The extensions to the DFM software implemented in Phase II include:

-- Capability to construct models in sub-modules

-- Automatic generation of decision tables from a set of rules

-- A library of cognitive modules

2.1.3 Phase II Technical Objective 3: Development of Software

The third technical objective of the Phase II research was to develop the software extensions according to the specifications developed as part of Technical Objective 2.

2.1.4 Phase II Technical Objective 4: Validation of Modeling Approach and Testing of Software

The fourth technical objective of Phase II was to validate the modeling approach finalized as part of Technical Objective 1, and to test the software that was developed as part of Technical Objective 3. To accomplish this technical objective, a case study taken from a real power plant application was used.

3. OVERVIEW OF DFM

The DFM approach (Garrett, et al., 1995a; Garrett, et al., 1995b; Guarro et. al., 1996; Yau, 1997) is based on representing the system under analysis with a "digraph" (directed graph) model. The digraph model explicitly identifies the cause-and-effect and timing relationships between the parameters and states that are best suited to describe the system behavior. Once such a model has been produced, automated deductive or inductive algorithms built into the methodology can be applied. The deductive procedures are applied to identify how system states -- which may represent specific success or failure conditions of interest -- can be produced by combinations and sequences of basic component states. Conversely, inductive procedures can be applied to the same model to determine how a particular combination of basic component states can produce various possible event sequences and subsequent system-level states. Thus, DFM can provide the multi-state and time-dependent equivalent of both fault tree analysis (FTA) and failure modes and effects analysis (FMEA), with the advantage that a single DFM system model contains all the information necessary for the automated execution of these analyses for practically any system condition of interest. This can be compared, for example, with the execution of FTA, in which each system "top event" requires a separate manual analysis and the construction of a separate fault tree model. A similar comparison can be made between FMEA and DFM. In performing a failure modes and effects analysis, the causality relationship in the system has to be revisited for each analysis to deduce the effects of different failure modes. In the DFM framework, on the other hand, once a model has been developed, the automatic inductive analysis algorithm can produce an entire array of separate automated analyses, to show how any initially hypothesized component failure may progress through the system, without further reasoning inputs from the analyst. Moreover, this inductive algorithm can even automatically handle cases in which the failure modes may branch into different areas of the system, with separate effects that recombine later in interactions further downstream in the flow of system cause and effect.

The application of DFM is typically a two-step process:

Step 1: Build a model of the system for which an inductive or deductive analysis is required. For applications in a human failure context, the model should encompass both the team of operators and the system being operated. The model expresses the functional and cause-effect relationships among the system physical and human variables, as well as the time dependent aspects of the system behavior.

Step 2: Using the model constructed in Step 1, either:

-- perform a deductive analysis to search for system and process failure states, in combination with human actions and states, that may occur as a result of the propagation through the system of perturbations produced by basic "root cause" events (an abnormal system condition); or

-- perform an inductive analysis to generate the sequence of events that will result from a specific set of initial and boundary conditions. This involves the identification of the effects (expressed in terms of the values of system and/or component states and/or process parameters), over a number of time steps that result from the set of initial conditions and boundary conditions;

In a deductive analysis, the system states for which the root causes are sought can be desirable or undesirable, depending on the objective of the analysis. The root causes are identified by backtracking through the DFM model of the system in a systematic, specified manner, and by expressing the conditions that cause the system events of interest in the form of timed prime implicants. The intermediate conditions identified along the backtracking process are summarized as intermediate transition tables. These intermediate transition tables can be represented also in the form of timed fault trees. The information contained in the fault trees that describe the conditions that can lead to system states of interest can be used to uncover undesirable or unanticipated human and system interactions.

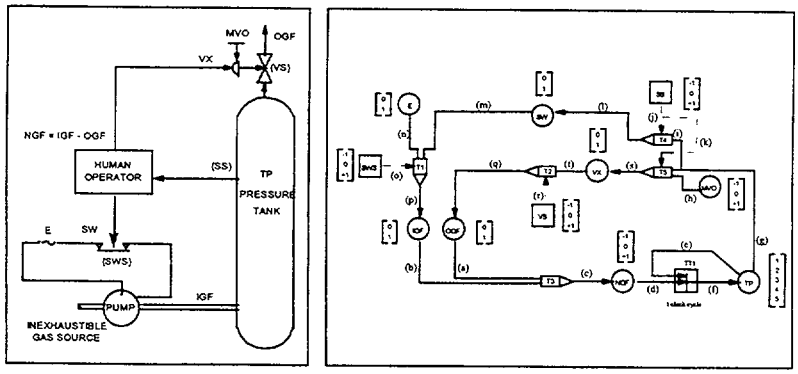
An inductive analysis, on the other hand, starts from an initial system condition and traces the DFM model forward in causality to identify the sequence of events that follows from the initial condition. This sequence is expressed in terms of the values of the DFM variables in each succeeding time step that follows the initial time step. The direct cause and effect relationships between the values of these DFM variables are summarized in the form of transition tables, which can be translated into a graphical format such as an event sequence diagram. The initial condition and

the boundary conditions can be defined to represent the normal or degraded system states. With the normal states, an inductive analysis can be carried out to verify that the system can indeed accomplish the design goals (verification analysis). With degraded system states, inductive analyses can be used to find out the effects of different combinations of failure modes, such as single failures or double failures, on the control functions being implemented in the system (automated FMEA).

It should be noted that, once a DFM system model is constructed, it can be used to analyze many different top events and many different combinations of sub-system failures; that is, the same model can be used repeatedly to check many different system conditions/states of interest. Thus, the time and resource investment associated with the construction of a DFM model has a high return, since, once this model has been built, the automated analysis engine can generate as many fault trees or event sequence diagrams as needed.

3.1 Framework for Model Construction (Step 1)

A DFM model expresses the logical and dynamic behavior of a generic system. A DFM model is an integration of a "time-transition network", a "causality network" and a "conditioning network", which is built by using detailed multi-state representations of the cause-and-effect and time-varying relationships that exist among the key system and human parameters. To illustrate to the reader how DFM modeling and analysis steps can be executed in a typical application, and anchor the DFM modeling concept and building blocks with an example, we present them in the following within the context of a simple system.



(a) System (b) DFM Model

Figure 3.1: A simple system and its DFM model

Figure 3.1 shows the schematic of a gas-storage system with its associated pressure control system, which we can assume, for the sake of discussion, to be implemented by a single human operator (note that, for the introductory nature of the discussion of DFM features that is sought in this chapter, it would make no real difference if we assumed the control system to be implemented by hardwired logic). Figure 3.1 also shows the DFM model of the integrated system.

The DFM model network is constructed from the DFM modeling elements. These modeling elements, as well as the manner in which they are assembled to form a DFM model, are discussed below.

3.1.1 DFM Modeling Elements

A DFM model makes use of certain basic modeling elements to represent the temporal relations and the logical relations that exist in the system and the associated software. More specifically, a DFM model integrates a "time-transition network" that describes the sequence in which human actions are carried out and the process evolves, a "causality network" that shows the functional relationships among key system states and process parameters, and a "conditioning network" which models discontinuous hardware performance due to component states and human performance parameters due to root causes of behavior. The building blocks of these three intertwined DFM

subnetworks are process variable nodes, condition nodes, causality edges, condition edges, and transfer and transition boxes with their associated decision tables. These basic modeling elements are shown in Figure 3.2

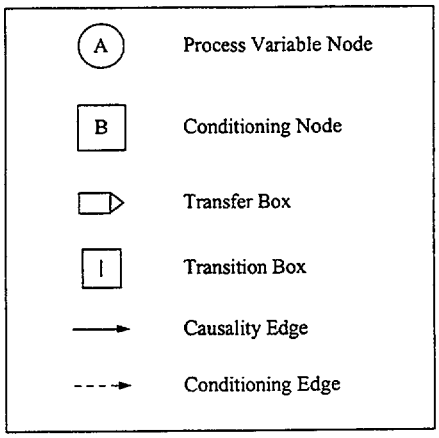


Figure 3.2: The basic DFM modeling elements

3.1.1.1 Process Variable Nodes

Process variable nodes represent physical and software variables necessary to capture the essential functional behavior, continuous or discrete, of the system and process. For example, the process variable node TP in Figure 3.1 represents the pressure in the gas tank.

A variable represented by a process variable node is discretized into a finite number of states. The reason for the discretization is to simplify the description of the relations between different variables. The choice of the states for a process variable node is often dictated by the logic of the system. For instance, it is natural to set a state boundary at a value that acts as a trigger point for a switching action or a value that indicates the system is progressing towards failure. The number of states for each variable must be chosen on the basis of the balance between the fidelity of the model and the complexity introduced by higher numbers of variable states.

For example, the process variable node TP in Figure 3.1 represents the tank pressure and it can vary from very low to very high. In our example TP is discretized into 5 states, and the discretization scheme of this process variable node is shown in Table 3.I. In this scheme state 1 signifies very low pressure (the tank is almost empty). State 2, state 3 and state 4 represent low pressure, normal pressure and high pressure respectively, while state 5 corresponds to dangerously high pressure, which can cause the tank to burst. The state boundary between 2 and 3 is set to correspond to the trigger point where gas inflow is activated to replenish the tank. Similarly, the boundary between states 3 and 4 corresponds to the set-point for opening the relief valve to reduce the pressure in the tank.

Table 3.I: Discretization scheme for the process variable node TP

State	Description
1	Tank pressure is very low
2	Tank pressure is low
3	Tank pressure is normal
4	Tank pressure is high
5	Tank pressure is very high

3.1.1.2 Causality Edges

Causality edges are used to connect process variable nodes to indicate the existence of a direct cause-and-effect relationship between the variables described by the nodes. For example, the causality edges (a), (b) and (c) in Figure 3.1(b) show that the value of the process variable NGF (net gas flow into the tank) is directly related to the

values of the process variables IGF (gas inflow into the tank) and OGF (gas outflow through the valve at the top of the tank). The precise nature of the functional relationship (i.e., the discrete state "transfer function") is described by a "transfer box" that is always directly associated with each causality edge (please see discussion in Section 3.2.2.1.3 below).

3.1.1.3 Transfer Boxes and Associated Decision Tables

A transfer box represents a transfer function between process variable nodes. The quantification of the transfer function, i.e., the correlation between the states of the input process variable nodes and those of the output process variable nodes, is actually described by decision tables associated with each transfer box.

Each transfer box decision table quantifies the relationships between the transfer box input and output process variable nodes. This table is a mapping between the possible combinations of the states of the input process variable nodes and the possible states of the output process variable nodes. Decision tables are extension of truth tables in that they allow each variable to be represented by any number of states. Decision tables have been used by other researchers to model components of engineering systems and construct binary fault trees (Salem, et. al., 1977; Salem, et. al., 1979; Ogunbiyi, 1980; Henley and Kumamoto, 1992).

Because each transfer box input or output variable is a vector of states, and each combination of input states maps to a state of each of the output variables, each decision table is actually a multi-dimensional matrix whose dimension is equal to one plus the number of its inputs. For simplicity and convenience of representation, all decision tables can be reduced to a two-dimensional form. In this simplified form, there will be a column for each input variable and a column for each output variable of interest. For example, in Figure 3.1(b), transfer box T3 links the input nodes IGF and OGF to the output node NGF. IGF is discretized into 2 states (0,1), as is the other input node OGF (0,1), while the output node NGF is discretized into 3 states (-1,0,+1). Hence in the decision table, there are 3 columns (1 for each of the two inputs and 1 for the output). The decision table in Table 3.II shows the output states produced from different combinations of the states of the inputs.

Table 3.II: Decision table for the transfer box T3 in Figure 3.1

IGF	OGF	NGF
0	0	0
0	1	-1
1	0	+1
1	1	0

Decision tables can be constructed from empirical knowledge of the system, from physical equations that govern the system behavior, and human failure events that result from UAs and EFCs. Decision tables that represent human interactions and cognitive behavior can be constructed from procedures and knowledge about cognitive behavior. This is discussed further in Section 5.4. To achieve a good fidelity, a judicious selection needs to be made of the number of states into which each node is discretized. The flip side of the accuracy requirement is the need to keep decision tables from growing too large, as this may lead to a combinatorial explosion of states to be tracked in the DFM deductive mode of analysis.

3.1.1.4 Condition Edges

Unlike causality edges, condition edges are mostly used to represent true discrete behavior in the system. They link parameter nodes to transfer boxes, indicating the possibility of using a different transfer function to map input variable states into output variable states. For example, as shown in Figure 3.1(b), the output OGF (gas outflow through the valve) can be proportional to the input VX (valve position), or be stuck at the minimum or maximum value regardless of VX, depending on the value of the parameter VS (unfaulted or faulted state of the valve).

3.1.1.5 Condition Nodes

Condition nodes, like process variable nodes, represent physical or human parameters. However, condition nodes are used in DFM to explicitly identify component failure states, changes of process operation regimes and modes, and root causes of human behavior. Condition nodes represent variables that can affect the system by modifying the causal relations between the basic process variable nodes. Some condition nodes may also be process variable

nodes themselves, linked by causality edges to other upstream process variable nodes, but condition nodes whose states are not determined by other upstream process variable nodes are treated in DFM as "random variables", i.e., as variables that can be assumed to be in any of their possible states. In the latter case, a distribution of "relative frequency" of the associated states could also be assumed, for purposes of probabilistic quantification. For example, node VS in Figure 3.1(b) is a condition node that is not affected by any upstream process, as the failure of the valve is assumed to be a random event and is not explicitly modeled. It should be noted that the effect of a condition node on an output variable is modeled through a decision table, as is the case for a process variable node. The reason for having the added modeling elements of condition nodes and condition edges is to offer a clear distinction between normal, off-normal or "modified" operation of a system.

3.1.1.6 Transition Boxes and Associated Decision Tables

Transition boxes are similar to transfer boxes in that they connect process variable nodes to indicate cause-and-effect relationships. Condition nodes can also be associated with transition boxes to represent modified behavior and relationships between the input and output process variable nodes. Decision tables are again used to describe the relationships between the input and output process variable nodes. However, transition boxes differ from transfer boxes in the essential aspect that a time lag or time transition is assumed to occur between the time when the input variable states become true and the time when the output variable state(s) associated with the inputs is (are) reached. This time delay is a characteristic of the transition which is being modeled and is treated as an attribute of the transition box. For example, in Figure 3.1, the transition box TT1 indicates that a new value of TP (an updated value of the tank pressure) depends on the value of NGF (the net gas flow into the tank) and the old value of TP (the tank pressure at the previous clock cycle). Transition boxes are routinely used in DFM to model the execution of software routines and the handling of interrupts, which often play an important role in the execution flow of digital control system software. They can, of course, also be used to model hardware time transitions.

3.1.2 Model Construction and Integration

To construct a DFM model for a digital control system, the first step is to select the physical components and the software functions that are to be included in the model. Following that, the physical parameters and software variables that capture the essential behavior of these components and software functions are identified and represented as process variable nodes. These process variable nodes are then linked together by causality edges through transfer boxes or transition boxes to form an integrated "causality" and "time-transition" network. Discrete behaviors such as component failures and logic switching actions are then identified and represented as condition nodes, which are tied to transfer boxes and transition boxes expressly to show how a "conditioning network" of discrete actions and events actually interacts with and affects the integrated "causality" and "time-transition" network. The parameters represented by the process variable nodes and condition nodes are discretized into meaningful states, and decision tables are constructed to relate these states. The decision tables can be constructed by empirical knowledge of the system, from physical equations that govern the system behavior, and from available operating procedures and knowledge (or assumptions) about human behavior. The completed DFM model then reflects the essential causal, temporal, and logical behavior of the entire human/physical system.

3.2 Framework for Model Analysis (Step 2)

The analysis of a DFM system model constructed according to the rules described above (Step 1) can be conducted by tracing sequences of events either backward from effects to causes (i.e., "deductively"), or forward from causes to effects (i.e., "inductively") through the model structure. This section presents the theoretical basis on which the DFM deductive and inductive analysis procedures are developed.

3.2.1 Similarities between DFM Deductive Analysis and Fault Tree Analysis

In a DFM deductive analysis, the goal is, starting from a defined system condition, to work backward in cause and effect flow to identify the paths and reverse sequence of events and conditions by which the system condition of interest (which in the DFM framework can itself be defined as a combination of events) may be produced. The conclusion of this process is to eventually identify in this fashion what sets of "basic" system events (i.e., combinations of basic hardware and software conditions) may be at the root of the hypothesized system condition. This kind of DFM analysis thus shares many of the conceptual features of fault tree analysis. A fault tree is a graphical model that represents the combinations of individual component failures which can lead to an overall system failure (referred to as the top event). In conventional binary fault tree analysis, once a fault tree has been

developed, application of Boolean algebra can reduce the tree to a logically equivalent mathematical form in terms of the tree's minimal cut sets. A cut set is defined as a set of events that, if they all occur, will lead to the top event. A minimal cut set is a cut set that does not contain any other cut set as a subset. The removal of any event from a minimal cut set would cause it to no longer be a cut set.

To illustrate the above in formal notation, let X_{top} be an indicator variable for the top event. An indicator variable can take the value of either 0 or 1 (0 if the top event is false, and 1 if it is true). Similarly, let $X_i^{(j)}$ be an indicator variable for the i-th primary event in the j-th minimal cut set. Then the indicator variable for the j-th minimal cut set, MCS_j, is a monomial that can be expressed as the conjunction of the indicator variables of its primary events:

$$MCS_j = \prod_{i=1}^{n} X_i^{(j)} \qquad \text{(Eq. 3.1)}$$

where n is the number of primary events in the j-th minimal cut set. The indicator variable for the top event can then be expressed in disjunctive form as:

$$X_{top} = 1 - \prod_{j=1}^{m} (1 - MCS_j) \qquad \text{(Eq. 3.2)}$$

If the variables that appear in a binary fault tree are appropriately defined, the formula that expresses the top event as a function of the basic events (equation (3.2)) exhibits coherent behavior. In particular, when a basic event variable changes from the value 0 to the value 1 (i.e., according to convention, from the unfaulted to the faulted state) the top event variable can remain at the value 0, or change from 0 to 1 (if it was at 0 before the basic event change), or remain at the value 1 (if it was already at 1 before the basic event change), but never go from 1 back to 0.

3.2.2 Multi-Valued Logic Trees and Prime Implicants

A fundamental limitation to conventional fault tree analysis is that the above method can only be applied to systems in which the primary events, $X_i^{(j)}$, are binary. Dynamic system modeling tends to require representation of physical variables (e.g., pressure, temperature and voltage.) and other complex system variables, thus binary logic (in which only two states may be used to characterize each variable) is, in general, not sufficient for an adequate representation of the behavior of the system. DFM models thus employ multi-valued logic (MVL), wherein each variable space may be discretized into an arbitrary number of states. A DFM "fault tree", therefore, would contain non-binary primary events (or certain equivalent binary expressions containing groups of mutually exclusive binary primary events which signify whether a given multi-valued variable is in a particular state). Although a definition of a coherent MVL tree can be given, most MVL trees of practical interest (and their equivalent binary expressions), including DFM-derived fault trees, are non-coherent. An intuitive, rather than formal, way of understanding this is by noting that DFM variable states are not ordered in such a way that higher states always indicate "increasingly-faulted" conditions and lower states always indicate "increasingly-nominal" conditions. Thus, as a basic variable changes from a lower to a higher state, the system-state indicator variable of choice for the particular analysis of interest may be going in the opposite direction, i.e., from a higher to a lower state.

The top event of a MVL fault tree can still be expressed in disjunctive form (the form of a disjunction of conjunctions of primary events), but the MVL analogue of the minimal cut sets encountered in binary fault trees are known as prime implicants (Henley and Kumamoto, 1992; Ogunbiyi, 1980; Ogunbiyi and Henley, 1981; Garriba, et al., 1985; Shields, et al., 1994). A prime implicant is any monomial (conjunction of primary events) that is sufficient to cause the top event, but does not contain any shorter conjunction of events that is sufficient to cause the top event. The prime implicants of a function are unique and finite (Quine, 1955); however, finding them is a more challenging task than finding binary logic minimal cut sets.

DFM uses decision tables to map the combinatorial states of transfer box inputs to their outputs. Decision tables allow each variable to be represented by any number of states, and they have been applied in fault tree analysis in the past to model component behavior. Given the state of a transfer box output node, the decision table gives the complete sets of inputs that could have caused it. Since a decision table is, itself, essentially a disjunction of conjunctions of states, it is possible to generate prime implicants from the table (Henley and Kumamoto, 1992). Methods have been developed for obtaining system prime implicants from component decision tables (Henley and Kumamoto, 1992; Ogunbiyi, 1980). The fundamental approach is to combine the individual component decision tables into a single critical transition table (Henley and Kumamoto, 1992; Kumamoto and Henley, 1979), and

perform a series of absorption and merging operations (Quine, 1952; Quine, 1955; Mott, 1960) on the rows of the table to reduce it to the complete set of prime implicants. In DFM, the procedure for generating prime implicants has been extended to carry out deductive analysis across time transitions, so that dynamic representations of systems can be analyzed.

When referring to prime implicants in the context of a DFM analysis, another important observation is that the presence of the time element in the DFM modeling framework introduces the possibility of prime implicants that would not be possible in ordinary time-invariant logic. In the latter, for instance, a prime implicant of the form:

$$\text{variable A} = 2 \ \wedge \ \text{variable A} = 3$$

would not be possible, and, if found in the course of a time-invariant analysis, would have to be eliminated by application of explicit "physical consistency rules". In the application of DFM to time-dependent systems however, if a time-transition has been encountered and the prime implicant is thus "time-stamped" to indicate:

$$(\text{variable A} = 2 \ @ \ \text{time t} = T1) \wedge (\text{variable A} = 3 \ @ \ \text{time t} = T2),$$

then the logical inconsistency no longer exists, and the prime implicant can be considered possible (unless of course it violates a "dynamic consistency rule", which still applies in time-dependent logic; please refer to Section 3.2.3.3). All prime implicants identified in a DFM analysis are conjunctions of primary events with associated time stamps, and they are simply referred to as "timed prime implicants" (TPI's).

DFM, therefore, represents a significant advancement beyond conventional fault tree analysis. In particular, a conventional fault-tree produces cut-sets for one, and only one, binary top event, with no associated time dependent information. The DFM representation is considerably more powerful, because it produces multi-valued logic and time-dependent prime implicants for a very large number of possible top-events. A DFM top-event can, in fact, be chosen to be any state among all the possible states of any of the variables, or even any combination of states of separate variables across time boundaries. As we have mentioned earlier, once a DFM system model has been constructed, it can be used repeatedly to investigate many different top events.

The algorithms for the identification of TPI's can produce different types of information, depending on the level of detail included in the original DFM model. More specifically, if the system is only modeled to the major component level, so that each system component or module is represented in DFM as a relatively high-level "transfer box" between "global" system-level principal input and output variables, then by definition the top-event prime implicants will only be expressed in terms of the states of such system-level variables (i.e., not in terms of variables that are "internal" to each software module). Another option in the type of information sought is whether the DFM backtracking is not only conducted, but also reported by the DFM software module by module and component by component, so that, when the process is completed, information equivalent to an actual "timed fault tree" (TFT) is produced as output of the analysis, along with its TPI's. It should be noted that, as discussed further in Section 3.2.3.1, the backtracking process is actually conducted step by step within the DFM algorithmic procedure. Therefore decision-table-format information, equivalent in substance to a timed fault tree, is produced as an intermediate result on the way to identifying the top-event TPI's and can be reported to the user. The timed fault tree, when read from the basic events to the top, provides the "explanation" and illustration of how, starting from the basic events contained in the prime implicants at the bottom of the tree, the system evolves through a sequence of states which finally lead to the top-event identified at the top of the tree. Note that the actual progression of cause and effect in the process is exactly in reverse order with respect to the order in which the DFM model analysis unravels the event-sequence, backward in causality and time, from the ultimate system-level effect down to the basic events that are at its origin.

3.2.3 Deductive Analysis Procedure

The following subsections discuss the analytical procedures employed in a deductive analysis.

3.2.3.1 Intermediate Transition Table and Timed Fault Tree (TFT) Construction

In the deductive analysis of a DFM system model, a particular system condition of interest (desirable or undesirable) is first identified. This system condition is expressed in terms of the state(s) of one or more process variable nodes,

which are thus taken to be the fault tree "top event(s)". The DFM model is then analyzed by backtracking, via a computerized analytical procedure, through the network of nodes, edges and transfer boxes, and through the time transition network which keeps track of timing effects. This "automated back-tracking procedure" is continued for a few steps back in time, producing along the way the intermediate transition tables associated with the particular top-event of interest, to find the possible "cause(s)" of that top event. The causes are expressed in terms of the combinations of the basic system variable states which may produce the top event. The order in which the transfer boxes are visited in reverse is dictated by the logical sequence of the boxes in the DFM model, as well as by the sequence of transitions (corresponding to the order of execution of the software modules or physical events associated with time delays) in the time-transition network. The information discovered at each step of the backtracking process is represented in the form of a series of intermediate transition tables, which are logically equivalent to gates in a timed fault tree.

To illustrate this analytical process, as it would be implemented in a manual execution, consider the analysis of the tank pressure control system shown in Figure 3.1. A top event has been defined as a situation in which the pressure in the tank reaches a dangerously high level. This top event is first translated into the state of the process variable node { TP = 5 @ t = 0 } and is represented in the transition table format in Table 3.III. The transition table shows that the TOP is true if TP is in state 5 at time 0. This table can be translated into timed fault tree gates as shown in Figure 3.3(a). This event is to be expanded by backtracking through the model. From the DFM model in Figure 3.1, TP at t = 0 is calculated from TP at t = -1 and NGF at t = -1 through the transfer function associated with the transition box TT1. The decision table for transition box TT1 is then consulted to identify combinations of TP and NGF at a previous time step that can cause TP = 5 at the current time step. In this case, the two events {TP = 5 @ t = -1} OR {(TP = 4 @ t = -1) AND (NGF = +1 @ t = -1)} are found to be the causes, and they are entered into a new intermediate transition table as in Table 3.IV. The equivalent timed fault tree is shown in Figure 3.3(b). Note that a dotted line separates the top event and the events at the second level to indicate the presence of a time transition between the events at the two different levels. Next we backtrack through transfer box T3, in the DFM model in Figure 3.1, to find the combinations of IGF and OGF which can cause NGF = +1. One combination is identified and is shown in Table 3.V and in Figure 3.3(c) as an AND gate joining the particular states of IGF and OGF. Backtracking through the transfer boxes T1 and T2 will give us the causes for IGF = 1 and OGF = 0, respectively. The backtracking steps are repeated to produce intermediate transition tables that are translated into the timed fault tree shown in Figure 3.4.

Table 3.III: Example of intermediate transition table construction

TP @t = 0	TOP
5	T

Table 3.IV: Example of intermediate transition table construction

NGF @t = -1	TP @t = -1	TOP
+1	4	T
-	5	T

Table 3.V: Example of intermediate transition table construction

OGF @t = -1	IGF @t = -1	TP @t = -1	TOP
0	1	4	T
-	-	5	T

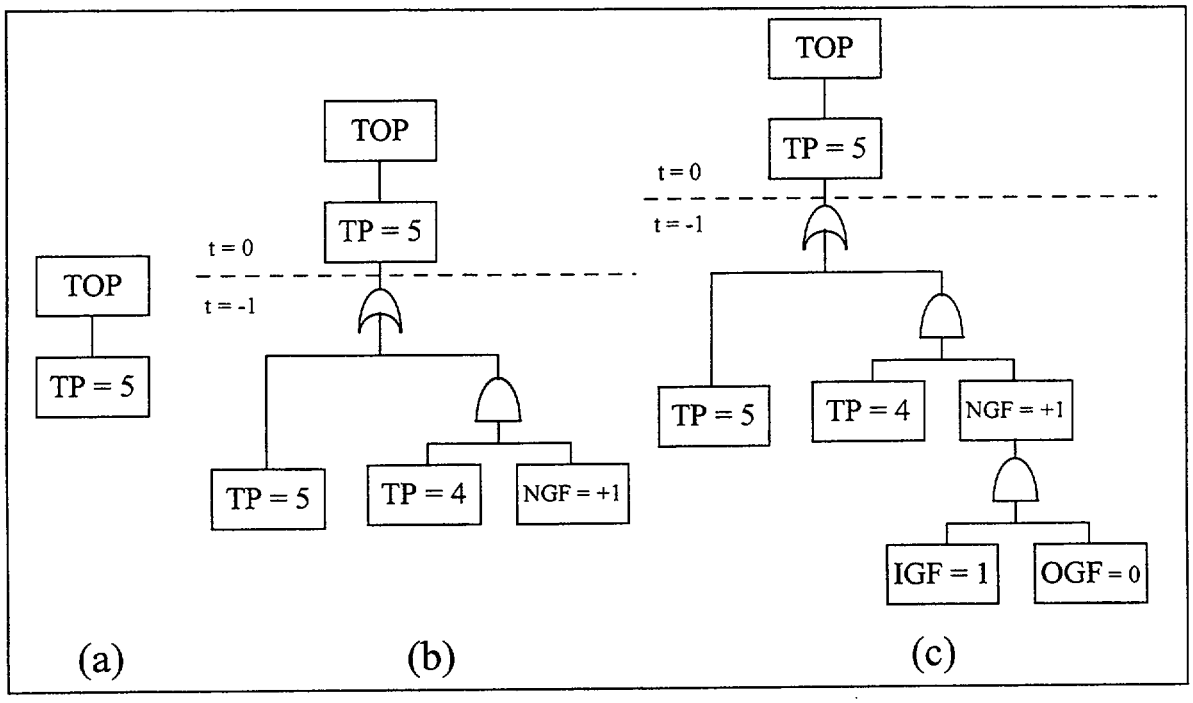


Figure 3.3: Example of timed fault tree construction

In many digital control systems, there are feedback or feedforward characteristics. This can cause a node to be traced back to itself in the fault tree construction. Consistency rules must be applied when these situations are encountered. Inconsistent branches are then pruned from the timed fault tree. Two major classes of consistency rules have been identified, and they are "physical" consistency rules and "dynamic" consistency rules.

3.2.3.2 Physical Consistency Rules

Physical consistency rules are applied to eliminate physically impossible conditions from the timed fault trees. An example of this would be a system parameter taking on two different values at the same time step in the timed fault tree. This class of consistency rule is similar to the consistency rules applied in conventional static fault tree analysis. If the same variable appears twice, but in different states, in the same time step and under the same AND gate, then everything beneath the first AND gate above the second occurrence of the event must be pruned from the tree due to physical inconsistency. This is illustrated in Figure 3.5(a). If pruning this AND gate causes events above to become impossible, then these events must be pruned as well. Such is the situation illustrated in Figure 3.5(b).

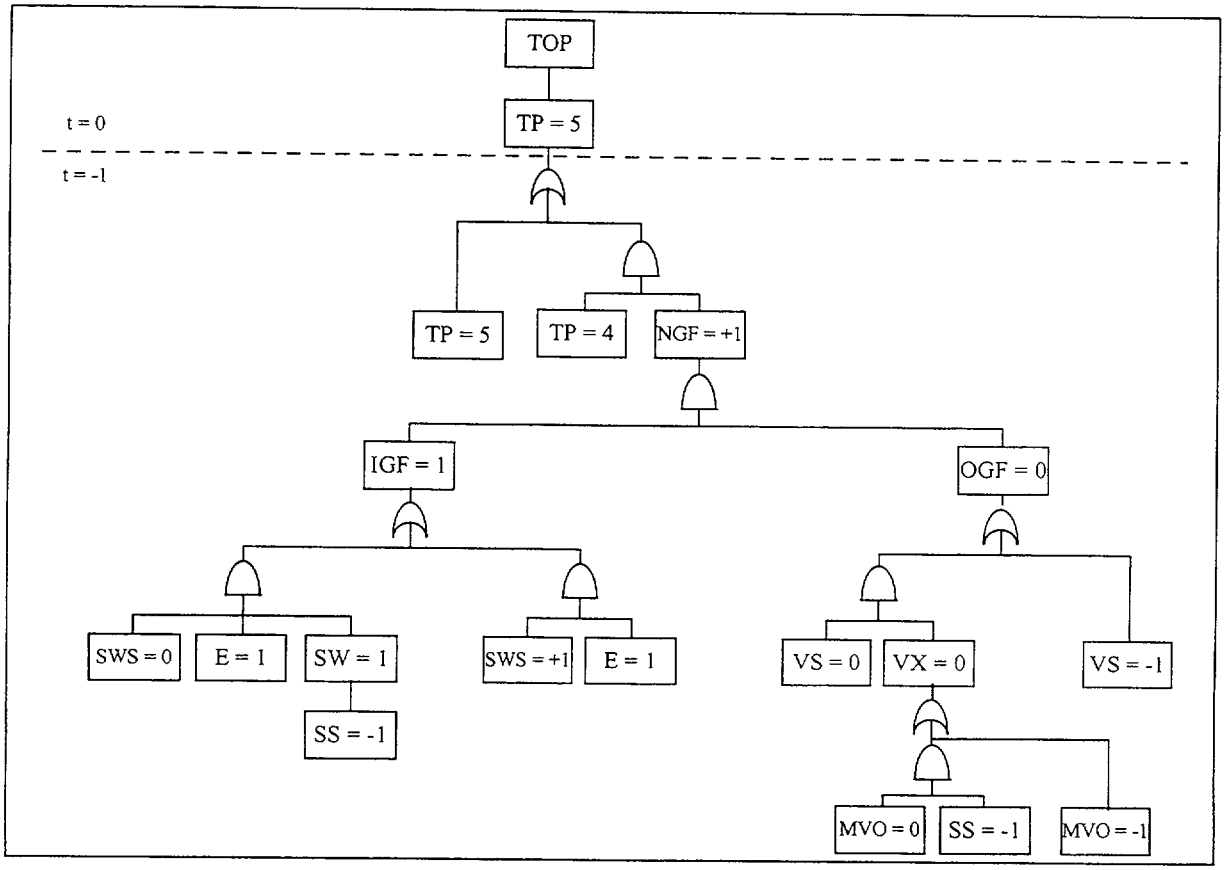
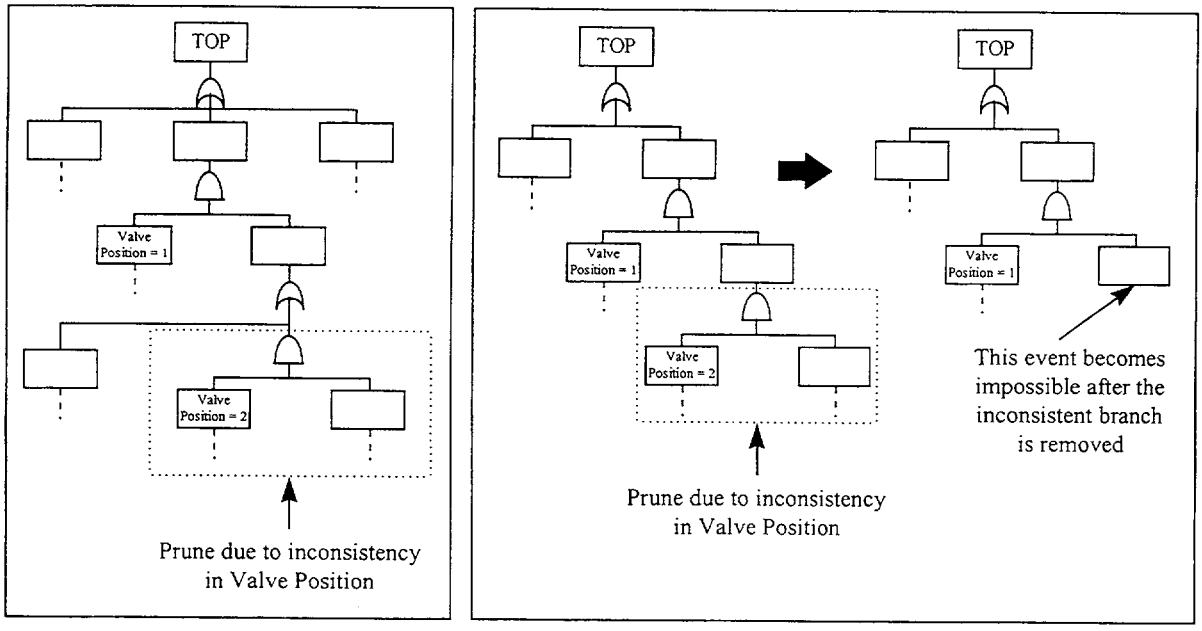


Figure 3.4: Timed fault tree for very high tank pressure



(a) (b)

Figure 3.5: Illustration of physical inconsistency

3.2.3.3 Dynamic Consistency Rules

Dynamic consistency rules, likewise, are applied to the timed fault trees to eliminate branches that violate the constraints on the dynamic behavior of the system under consideration. These rules are developed from the analyst's knowledge and assumptions about the system's dynamic behavior. Dynamic consistency rules are expressed in terms of allowable variations of parameter values across different time steps. Some possible forms of dynamic consistency rules are:

-- The state of a parameter cannot change in a certain direction between two time steps.
-- A parameter cannot change by more than a certain amount of states between time steps.
-- Several parameters must vary in a specific way between two time steps.

Rules of the first type can be defined from the analyst's knowledge about the dynamic constraints of the system. For instance, in modeling a drain tank system, the level in the tank cannot increase with time if inventory is constantly being used up and is not being replenished. Rules of this type can also come from modeling assumptions. For example, if the analyst assumes the equipment in the tank system can only fail permanently, then a failed valve cannot return to the normal state in a later time step.

Rules of the second and third types come from knowledge of the system. For instance, a rule of the second type can state that the position of the valve cannot vary by more than two states in one time step, as it takes a finite amount of time for the valve to open or close. Similarly, an example of a rule of the third type can be the constraint that the valve position and flowrate must vary in a proportional manner as required by physical law.

Caution must be applied when defining dynamic consistency rules for a DFM failure analysis, in that the analyst must be sure that, by "enforcing" a rule, an abnormal but possible type of behavior is not ruled out. For example, in the tank pressure example used throughout this discussion, one may define the dynamic rule that the tank pressure cannot go from "normal" to "very low" in one time step, if it assumed that the gas outflow can only occur through the relief valve. However, this dynamic rule no longer makes sense, if an explosion and semi-instantaneous decompression of the tank is one of the possible scenarios that the analyst is interested in investigating.

Dynamically inconsistent branches are pruned in a manner similar to physically inconsistent branches. If a dynamically inconsistent event occurs in a timed fault tree, this event, including all of the sub-branches connected to it via the first parent AND gate, must be pruned. This is illustrated in Figure 3.6. As with physical consistency rules, further pruning may be necessary if branches that are eliminated can cause other events to become impossible.

3.2.3.4 Timed Prime Implicant (TPI) Identification

As discussed above, TPI's may be identified directly from a system DFM model. In the DFM analytical algorithm, decision tables encountered during the backtracking process are expanded and joined, one by one, to form a single critical transition table, which directly contains all of the system parameter states that are produced along the sequence leading to the top event. As mentioned earlier, in Section 3.2.3.1, the process of expanding and joining the decision tables in the backtracking process is logically equivalent to generating a timed fault tree, except that the events are not presented graphically as a tree structure, but in tabular form as intermediate transition tables. The critical transition table, on the other hand, is logically equivalent to the basic events produced in a timed fault tree. The reader should note that for a multi-state representation, the basic events identified in a timed fault tree (or the rows in a critical transition table) are the sufficient conditions for the top event. The complete set of unique timed prime implicants are produced by performing a series of absorption and merging procedures on the rows of the critical transition table, to reduce it to an irredundant form (Quine, 1952; Quine, 1955; Mott, 1960). For example, consider the decision table in Table 3.IV, which is the equivalent of a sum-of-products expression for some function, called TOP. The variables are assumed to be multi-state and their states are:

$A \in \{ -1, 0, +1 \}$,
$B \in \{ N, R, F \}$,
$C \in \{ -2, -1, 0, +1 \}$,
$D \in \{ H, N, L \}$.

(These variables and the corresponding decision table do not necessarily reflect any particular logic, but are merely intended to illustrate Quine's consensus operation.)

Table 3.VI: Decision table for function TOP

ROW	A	B	C	D	TOP
1	-	R	-1	N	T
2	0	-	+1	H	T
3	-	R	0	-	T
4	-	-	-1	L	T
5	0	R	-1	H	T
6	-	N	-2	-	T
7	-1	R	-1	H	T
8	0	R	-2	H	T
9	1	R	-1	H	T
10	0	F	-	H	T

In the application of the consensus operation procedure for Table 3.VI, rows 7 and 9 merge with row 5, yielding a "don't care" (which is represented by a "-") in column 1 of row 5 and a new decision table (Table 3.VII).

Table 3.VII: Decision table for TOP after merging operation

ROW	A	B	C	D	TOP
1	-	R	-1	N	T
2	0	-	+1	H	T
3	-	R	0	-	T
4	-	-	-1	L	T
5	-	R	-1	H	T
6	-	N	-2	-	T
7	0	R	-2	H	T
8	0	F	-	H	T

Rows 6-8 of Table 3.VII can then undergo a reduction operation, yielding a "don't care" in column 2 of row 7. Rows 1, 4 and 5 of the table also undergo a reduction-merging operation, yielding Table 3.VIII.

Table 3.VIII: Irredundant form of decision table for function TOP

ROW	A	B	C	D	TOP
1	-	R	-1	-	T
2	0	-	+1	H	T
3	-	R	0	-	T
4	-	-	-1	L	T
5	-	N	-2	-	T
6	0	-	-2	H	T
7	0	F	-	H	T

Rows 1-3 and 6 of Table 3.VIII yield a consensus term which is given in row 8 of Table 3.IX. Table 3.IX contains all of the prime implicants of the function since no new consensus terms can be generated from it and none of its terms can be simplified any further.

Of course, physical and dynamic consistency rules must still be applied during the construction of the critical transition table. The only difference is that, instead of applying them to individual events in the timed fault tree, they are applied to entire rows in the critical transition table.

Table 3.IX: Decision table for function TOP after consensus

ROW	A	B	C	D	TOP
1	-	R	-1	-	T
2	0	-	+1	H	T
3	-	R	0	-	T
4	-	-	-1	L	T
5	-	N	-2	-	T
6	0	-	-2	H	T
7	0	F	-	H	T
8	0	R	-	H	T

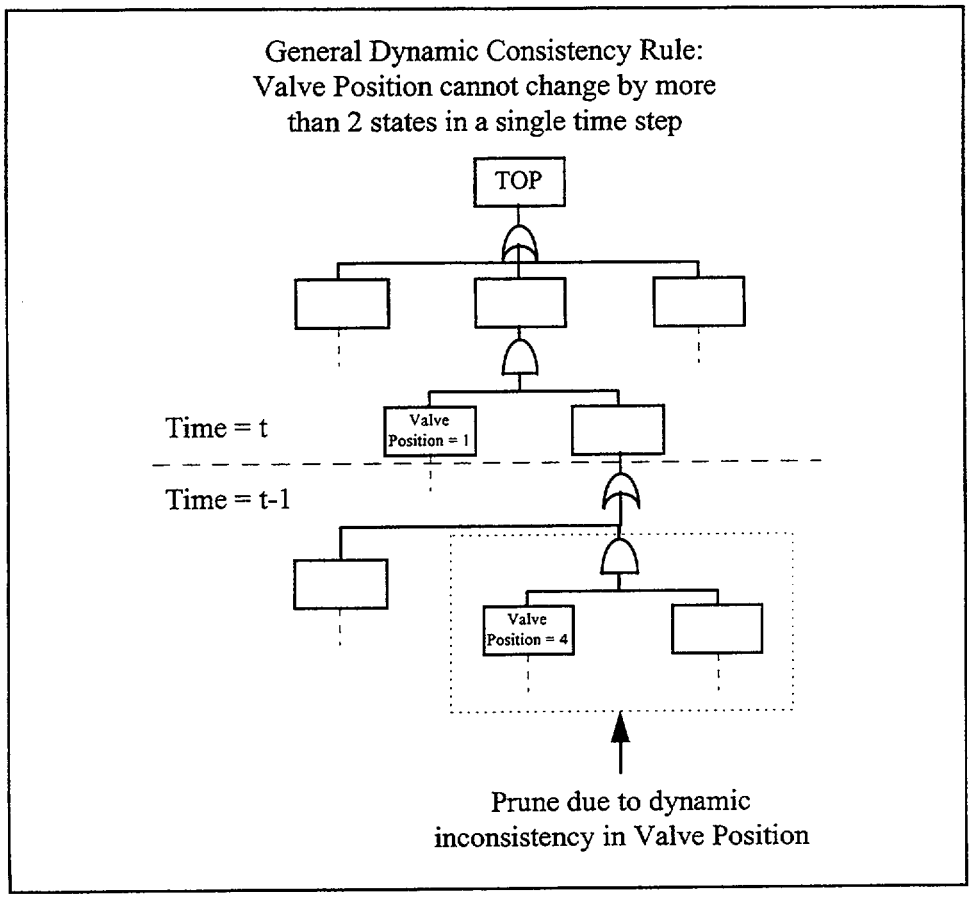


Figure 3.6: Illustration of dynamic inconsistency

It should be noted that the presence of "Don't Care" states can complicate the implementation of these dynamic consistency rules. In checking for dynamic consistency, the "Don't Care" state for a parameter might need to be replaced by a subset of the states for that parameter. A variable taking on the "Don't Care" state means that the variable can be in any one of its states. However, the presence of dynamic constraints for that variable reduces the domain of the allowable states. Hence, if a "Don't Care" state is encountered for such a variable, that "Don't Care" state must be replaced by the corresponding states allowed by the dynamic constraints.

3.2.3.5 Reduction of Prime Implicants

Prime implicants identified in a DFM analysis are expressed in terms of the states of the DFM model variables. As some variables represent the state of components, the prime implicants may contain non-failure conditions of these components. For ease of interpretation, a "reduced form" of the prime implicants can be obtained by deleting from

the list of conditions in the prime implicants, all those conditions which identify the states of sensors, control valves and stop valves related to the event sequence of interest as being normal, i.e. none of these components are failed.

In general, in a multi-state, non-coherent system representation such as that used in DFM, a parameter state can always be classified as "faulted" or "normal" only for the model parameters which are expressly intended to represent hardware failure and non-failure states. A reduced form of prime implicant can thus be obtained by not including in it the listing of normal states of this type of parameter. The states of process variables, on the other hand, are not definable, a priori, as being always "good" or "bad", and consequently are always listed, even in the reduced prime implicant. This is because a process parameter state which is "good" in a certain type of situation may become "bad" when the situation changes. The "goodness" of these process parameters cannot be determined until the context within which this happens has been identified.

3.2.4 Inductive Analysis Procedure

The following subsections discuss the analytical procedures employed in a inductive analysis.

3.2.4.1 Definition of the Initial Condition for an Inductive Analysis

A DFM inductive analysis starts from an initial condition, and traces the model forward in causality to identify the subsequent events that result from propagation through the system. The initial condition of an inductive analysis defines the status of the system for which the subsequent effects are of interest, and is expressed in terms of the states of the nodes in the DFM model. It is not necessary to specify the states for all the nodes initially, as some of these nodes are outputs of transfer boxes and can be calculated from their predecessor nodes. In fact, it is sufficient to specify, in the initial condition, the states of all the nodes that are either:

-- an output of a transition box, or
-- not an output of any transfer box or transition box.

The state of the output node for a transition box is calculated from the inputs of that box, which are expressed in terms of the states of the input nodes at the time step before the initiating time. To avoid going back recursively an infinite number of time steps to specify the initial state of the system, it is convenient to specify the state of such a node initially. For the DFM model shown in Figure 3.1, an initial condition for an inductive analysis must include the state of TP (the value of the tank pressure) at the starting time.

On the other hand, those nodes that are not outputs of any transfer box or transition box cannot be calculated from other nodes, as they do not have any predecessor. Hence, they must all be specified initially for the inductive analysis. This class of nodes usually represent basic components whose failures are assumed to be independent of the physical process represented functionally in the DFM model. For instance, to complete the definition of an initial condition for the DFM model shown in Figure 3.1, the initial states of SS (the pressure sensor), VS (the relief valve), SWS (electric switch), E (power supply) and MVO (operator override command) must all be specified.

For example, to investigate the effects of the relief valve failing closed when the pressure is above normal in an inductive analysis, the initial condition shown in Table 3.X can be used.

Table 3.X: Example of an initial condition for an inductive analysis

(TP = 4) ∧	Tank pressure is high AND
(VS = -1) ∧	Relief valve failed stuck closed AND
(SS = 0) ∧	Pressure sensor is normal AND
(SWS = 0) ∧	Electric switch is good AND
(E = 1) ∧	Power supply is available AND
(MVO = 0)	Operator does not command an override

3.2.4.2 Definition of the Boundary Conditions

In addition to defining the initial conditions, boundary conditions may have to be specified for the system if the inductive analysis is to be carried forward beyond the initial time step. In particular, if there are nodes in the DFM model which are not outputs to any transfer box or transition box, the states for these nodes must be specified

explicitly in all the future time steps covered in the inductive analysis. The explicit designation of their future states is required because these nodes have no predecessors upstream, and cannot be calculated from other nodes. These nodes form a subset of the nodes for which initial values must be specified, and are those nodes that are used to represent basic components in the model. When applied to a system design, the boundary conditions for an inductive analysis can be defined to reflect the failure profile of the components, so as to investigate the effects of different basic component failure combinations on the system itself. For instance, some components can be assumed to remain in their normal states throughout the analysis, while other components can be assumed to degrade from the normal state to a failure state at some future time step.

For example, to analyze the DFM model in Figure 3.1 inductively for 2 time steps (spanning time=0 and time=1), the states for the nodes SS, SWS, VX, E and MVO must be specified explicitly for time=1, as the values of these nodes at time = 1 cannot be propagated from the initial condition defined in Table 3.VIII.

3.2.4.3 Forward Propagation in an Inductive Analysis

After the initial condition and the boundary conditions have been specified, the inductive analysis can proceed automatically, requiring no further reasoning input from the analyst. In the first step of a DFM inductive analysis, the node immediately downstream of the nodes whose states are specified in the initial condition is evaluated. The evaluation of this intermediate node allows other nodes further downstream to be evaluated, in turn, and thus the inductive analysis is able to propagate from the initial condition through the system, producing values of intermediate nodes in the process. The evaluation of an intermediate node can be recorded in the form of a transition table.

For example, starting from the initial condition defined in Table 3.VIII, the nodes evaluated in the first step of the inductive analysis are VX and SW. In particular, TP=4 $\wedge$ MVO=0 $\wedge$ SS=0 result in VX=1 and TP=4 $\wedge$ SS=0 result in SW=0. This information can be summarized in the transition tables shown in Figure 3.7.

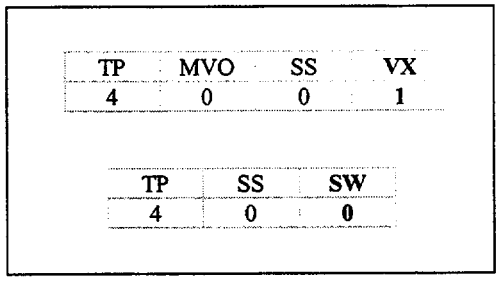


Figure 3.7: Transition tables showing the forward propagation

4. REVIEW OF HUMAN ERROR ANALYSIS METHODOLOGIES

Human error methodological structures were reviewed, as well as their model implementations (if developed), with the objective of defining a model structure for this DFM application. The resulting DFM methodology that uses the ATHEANA methodological framework, is presented in Chapter 5. The reasons behind the selection of the ATHEANA framework are discussed below.

4.1 Overview OF A Technique for Human Error Analysis (ATHEANA)

In this sub-section we present a review of recent advances in ATHEANA. This methodology was used as a basis for the development of much of the framework for this project because it includes consideration of team related events and errors of commission as well as errors of omission. ATHEANA provides an approach to the analysis of errors of commission and includes the consideration of human-system interactions. The framework brings together the disciplines of behavioral sciences, cognitive psychology and systems analysis. The ATHEANA application process is discussed in detail in (Cooper, et. al., 1996). Figure 4.1 shows the general structure of the process, and depicts five tasks for the analyst to perform, after the accident scenario has been defined. These tasks are from Parry, et. al., 1996:

-- Identification of the candidate human failure events (HFEs) to be modeled;

-- Identification of potentially important types of unsafe actions that could cause each HFE;

-- For each type of unsafe action, identification of the most significant reasons for that type of unsafe action and its associated reason, identification of the potentially significant error-forcing contexts (EFCs);

-- For each type of unsafe action and its associated reason, estimate the likelihood of the EFCs and the consequential probabilities of the unsafe actions; and

-- For each HFE, sum the likelihood of the EFCs and consequential probabilities of the unsafe actions for all potentially important types of unsafe actions that could cause the HFE.

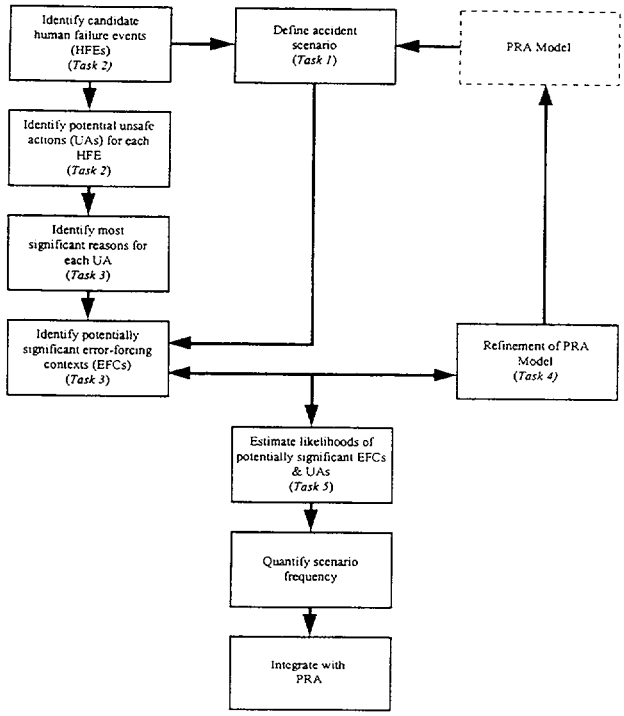


Figure 4.1: Process flow diagram (Parry, et.al., 1996)

The HFEs identified by the ATHEANA process correspond to human-caused failures at the function, system or component level and are usually correlated with hardware and software failure events that are already identified in the event trees, although it is possible that new failure events may be uncovered that are unrelated to any failure events in the existing event trees. It should be noted that software failure events are not typically modeled in a PRA. The application of the DFM modeling approach to tem related human errors of omission and commission has been developed within the framework of the definitions and their relationships that have been defined in the ATHEANA process. The application is consistent with the NUREG-1624 Draft Report, "Technical Basis and Implementation Guidelines for a Technique for Human Event Analysis (ATHEANA)", May, 1998. The DFM application includes the tasks that are defined in Section 2.2 of that report, including the definition of HFEs into the logic model, screening analysis, and the documentation of the process and the results. However the DFM application has not advanced to the quantification stage at this point in time, but it could be used for quantification after some future developments are completed.

The ATHEANA process provides "reasons" for unsafe actions that are based on a synthesis of psychological and plant factors that can lead the user to specific EFCs. Figure 4.2 shows the levels of "reasons" that are recommended for investigation after candidate HFEs and unsafe actions are identified. Inter and intra team interactions can be present at the information processing model stage relative to situation assessment, response planning, response implementation and monitoring.

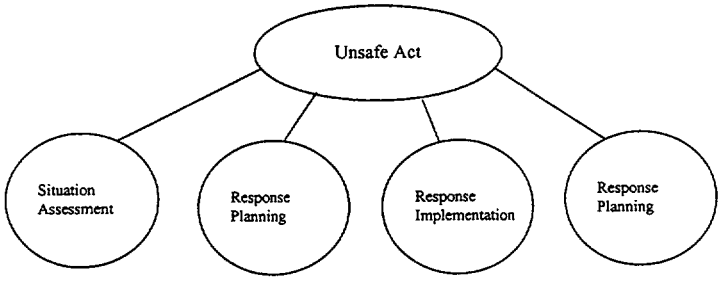


Figure 4.2: Information processing model in ATHEANA (Cooper, et.al., 1996)

4.2 Review of Other Dynamic HRA Methodologies

A discussion of dynamic approaches to HRA is presented in (Cacciabue, 1996). A methodology named HERMES (Human Error Reliability Methodology for Event Sequences) is discussed. HERMES focuses on the analysis of human interaction in dynamic conditions and its building blocks are models of plant, working environments and operator interactions. It is a simulation model that is based on the following items:

1. A definition of working environments, system designs including control systems as well as a definition of possible hardware failure modes
2. A model of functional response of the systems based on differential equations.
3. A cognitive simulation model which includes operator dynamic interactions with the plant. Operator cognition models represent the alternative operator responses to these interactions that include human error.
4. A classification of operator(s) errors that are included within the cognitive simulation model.
5. A method for structuring the interaction between the models of plant and cognition for controlling the dynamic evolution of events.

As stated in the reference document, HERMES is well suited for detailed operator and plant interactions analysis but care must be taken in setting up the simulation environment to control the explosive expansion of sequences generated by the simulation method. The DFM modeling framework is a method that inherently controls the number of sequences because it is based on a definition of a finite number of states that are correlated with the system models. It should be noted that the DFM application that uses the ATHEANA framework provides a basis for implementing Items 1,3 and 5 above.

Another dynamic simulation approach, called OPSIM, is discussed in Dang and Siu (1996). This model uses an operator-plant simulation. A cognitive model of the operator uses a blackboard architecture, and the overall simulation is implemented in C++ within a discrete event simulation framework. OPSIM emphasizes operator behavior based on schemas, and rules in combination with procedure following. In the schemas, cues from the plant, such as indications and alarms, are associated with plant states, system or equipment states. Secondly, schemas associate operator correct responses with these states. Errors can be modeled at the level of the knowledge base to reflect erroneous mental models. There are some features of this model that have similarities with the DFM methodology, but its application to team analysis raises the same issue of complexity as discussed above relative to the HERMES simulation model.

The dynamic event tree analysis method (DETAM) has been used in a realistic analysis to treat context, cognition and crew performance (Gertman, et. al., 1996). This approach, in which the dynamic evolution of possible scenarios is modeled explicitly, allows the treatment of various crew states (both perceptual and cognitive and psycho-social) and their interaction with the different plant states (as defined by key process variables as well as hardware status). In this sense the approach is similar to the DFM approach, but it does not offer the flexibility to screen the accident sequences in a PRA and to revise them as efficiently as can be accomplished with the DFM approach. The DETAM approach also allows the treatment of various performance shaping factors (PSFs) within the physical, cognitive and

psychological context of the evolving scenario. Unlike ATHEANA, it is not a logic structural framework, but rather an implementation methodology that forms a basis of comparison for the DFM implementation approach.

4.3 Cognitive Behavior of Operators and Root Causes of Errors

4.3.1 Overview of Cognitive Activities

A crucial aspect of the framework being developed here is the dynamic modeling of the cognitive activities associated with the four stages of information processing: monitoring and detection, situation assessment, response planning and response implementation. A particularly important issue is the identification and modeling of the root causes of cognitive errors for each of the four stages.

4.3.1.1 Situation Assessment

Situation assessment is the activity of constructing a mental model of the plant state based on observations. The operators update their mental model based upon information received about the plant state, both in terms of system states and process conditions. This information is generally received, either directly or indirectly, through instrument readings or indications, resulting from changes in the plant state or process behavior due to the progression of the accident or due to operator actions. The operators' situation assessment then guides their development of future response plans and further monitoring activities.

The activity of situation assessment is crucial to the development of EFCs and resulting unsafe actions. Unsafe actions such as mistakes occur because the operators have the wrong picture of what is actually going on. In other words, in the process of making a mistake, the operators inadvertently perform an action or sequence of actions that ultimately results in an HFE. However, their intent is correct based on their perception of the plant condition and behavior; it is their assessment of the situation that is wrong. Therefore, the activity of situation assessment, and errors in situation assessment leading to mistake- induced unsafe actions, will play a crucial role in the modeling framework being developed here.

Root causes of errors in situation assessment can be classified as an error in information collection, in which the operator receives the wrong information, or an error in information processing, in which the operator receives the correct information but still arrives at an incorrect assessment of the situation.

Root causes of errors in information collection include instrument failures or malfunctions, which can mislead operators into a wrong situation assessment without necessarily having a human, "cognitive", failure or error. Human-related failures in information collection include the operator reading the instrument wrong, or the operator not reading the instrument at all. The latter type of error can be broken down further: the instrument may be obscured from the operator's view, or the operator may not look at the instrument at all, due to a number of reasons, including inattentiveness and overload, etc. Errors in information collection can also be caused by a communication error, if the operator is receiving the information from another person.

Root causes of errors in information processing include cognitive biases, in which a cognitive bias allows the operator to believe something different than what he would if he had no bias, and an incorrect mental model of the behavior of the plant or process.

4.3.1.2 Monitoring and Detection

Monitoring and detection activities are either directed by procedures and the operators' mental model of the situation, or by alarms or other signals that get the operator's attention. Operator's responses to detectors (correct or incorrect) is taken into account. Monitoring refers to the operators' observance of the effects of their actions on the plant, and is explicitly taken into account. Thus, monitoring activities are directly linked to situation assessment, in that if the operators observe something different than what they expect, either they will change their mental model, or will disregard their observation.

4.3.1.3 Response Planning

Response planning is the process of deciding what actions to take. The operators' mental model of the plant state and the available procedures are used to formulate a response plan. One type of unsafe action related to the response

planning activity is a circumvention, which can happen if the operators decide that a step in the procedures is inappropriate or unnecessary based on their assessment of the situation. Another type of error related to response planning that can occur is if the operators implement the wrong procedures based on their understanding of the situation. After a response plan has been implemented, the operators will evaluate the effectiveness of their actions through monitoring and situation assessment activities.

Root causes of errors in response planning, if the crew is following written procedures, can include reading the procedures wrong or intentionally skipping or disregarding the procedures. If the crew is not following written procedures, then root causes of errors in response planning would primarily be due to incomplete or incorrect knowledge.

4.3.1.4 Response Implementation

Response implementation refers to the actual physical implementation of the response plan. Slips and lapses are generally the root causes of errors associated with response implementation. In general, unless an error in monitoring occurs, such errors are usually recognized and corrected, unless other PSFs are in effect, such as a fast-paced scenario or inattentiveness due to high (or low) workload. Communication breakdowns could also play a role in the team of operators not recognizing and recovering from a slip or lapse in response implementation.

4.4 Communication Errors

Figure 4.3 shows a model of communication between two persons. The communication process can be broken down into two sets of behaviors: creating and sending messages, and receiving and interpreting them (Barnes, et. al., 1996). To achieve a successful communication, the sender must compose a message that conveys the meaning as clearly as possible. The receiver must then receive the message and correctly interpret it. If the communication is successful, then the receiver will construct a meaning of the message that is similar to what the sender intended. The success of the communication can be verified through feedback, either in the form of additional communications, or through the observance of the effects of actions taken as a result of the message.

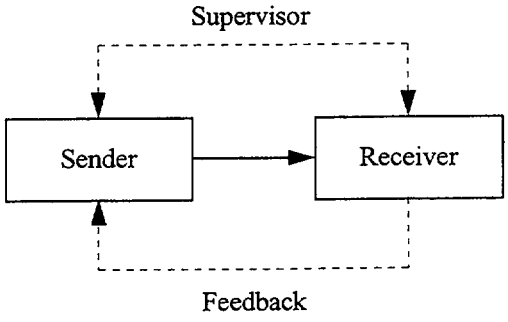


Figure 4.3: Communication model (Barnes, et. al., 1996)

Based on this model, Barnes, et. al., identify eight types of sending errors and four types of receiving errors. The sending errors include:

-- Message content is wrong. Communication fails because the information contained in the message is incorrect.

-- Message content is inconsistent with other information. The information in the message is correct, but is inconsistent with other information available to the receiver.

-- Message content is inappropriate for the receiver. The sender fails to tailor the message for the receiver in terms of the receiver's work context, the receiver's role in the task at hand, the receiver's technical knowledge, or fails to use terminology familiar to the receiver.

-- Message production is inadequate. After an accurate and complete message has been composed and tailored to the needs of the receiver, communication can fail if the message is not produced adequately.

-- Message is not sent. In this case the sender fails to transmit a message needed by the receiver.

-- Message is sent to the wrong place or person.

-- Message is sent at the wrong time. Communication fails if the sender transmits a message either too early or too late.

-- Failure to verify message understanding. The sender fails to take positive actions to verify that the intended receiver has accepted and understood the message.

The four receiving errors include:

-- Message is not sought. If the receiver does not actively seek the information necessary to perform a task then a receiver error may occur.

-- Message is not found or not used. In this case the receiver does not find (in the case of written communication) or disregards the message.

-- Message is misunderstood.

-- Receiver does not verify message understanding. The receiver fails to take action to test his understanding of the message received.

Some of the above messages are related to cognitive activities and should be modeled as such. Sender error 2, in which the message is correct but is inconsistent with other information available to the receiver, is similar to the situation in which an operator has conflicting or inconsistent instrument readings. As such, it should not be considered a communication failure, but should be considered as a context in which the receiver can make a cognitive error in situation assessment or response planning. With respect to the receiver errors, receiver error 1, in which the message is not sought, could also be considered an error in response planning that causes the communication failure. Receiver error 2, in which the receiver disregards the message, is also actually an error in response planning, in that the root cause of the failure is not part of the communication process.

5. DESCRIPTION OF METHODOLOGY

5.1 Overview

The DFM modeling and analysis framework focuses on the identification of unsafe actions and error forcing contexts that can lead to a human failure event in a team setting. The identification of unsafe actions and associated error-forcing contexts constitutes one of the most difficult tasks in the application of general human reliability methodologies, such as the ATHEANA approach. Thus, providing the technical means to achieve that objective also provides very useful support for the application of these general methodologies.

The DFM team assessment framework includes the development of an integrated DFM model, that explicitly considers the plant systems (including instrumentation) and processes, human behavioral modeling elements, and team factors., . In this way, possible error forcing contexts, as related to plant conditions, are included in the modeling framework. The framework also includes an analysis that identifies "prime implicants" which are the set of necessary and sufficient conditions that can lead to the top event, which in this case is the human failure event. Each prime implicant therefore will consist of a combination of states of the plant and one or more "unsafe actions" performed by the team, as conditioned by behavioral factors. These prime implicants can be used to identify unsafe actions and the error forcing contexts that trigger them.

This chapter describes the team assessment framework and the process that it supports for the identification of unsafe actions and associated error forcing contexts that may lead to a given a human failure event (HFE). The concepts of a human failure event, an unsafe action and an error forcing context are discussed in detail in the following paragraphs.

Human failure events (HFEs) can result in the loss of a function, as well as the associated systems or components. An HFE is caused by an unsafe action or sequence of actions that results in a worsened plant condition. An HFE can be either an error of commission (EOC), in which the operators disable or terminate a necessary safety function or initiate an inappropriate system, or with an error of omission, in which the operators fail to initiate a required safety system or function. Note that an HFE implies failure to recognize and recover from the error in time to prevent the plant's transition to a degraded state. In the context of PRA, in order to quantify an HFE it is necessary to identify the potential underlying unsafe actions and their associated error forcing contexts. Thus, the framework being developed here consists of step-by-step process of modeling and analysis for the identification of the unsafe actions and error forcing contexts that can lead to a specified human failure event.

An unsafe action represents an action, or sequence of actions, inappropriately taken, or not taken when needed, by plant personnel that result in an HFE. Unsafe actions can be classified according to a simple taxonomy developed by Reason (1990). This classification of unsafe actions include slips and lapses, mistakes and circumventions.

Slips and lapses are unsafe actions in which the outcome of the action is not what the person performing the action intended. They are generally associated with routine and highly practiced actions, and are often easily recognized and corrected.

Mistakes and circumventions are unsafe actions in which the action was intended. Mistakes are intentional actions in which the intention is wrong. A mistake can be considered to be "rule-based" or "knowledge-based". In a rule based mistake, specific documented instructions or procedures are being followed, and the mistake occurs because the rules or procedures are inappropriate for the actual situation (although they may be appropriate for the perceived situation). A knowledge based mistake occurs when there is no procedural guidance and the operator is relying on technical or specialist knowledge

Circumventions are intended unsafe actions in which an operator decides to break or skip some rule for what seems like a good reason. This intention could be based on the perception that the circumvention would have little or no impact on plant safety.

An error-forcing context (EFC) represents the combined effect of performance shaping factors (PSFs) and plant conditions that create a situation in which an unsafe act becomes likely (or possible). Performance shaping factors include conditioning elements such as workload, stress, poor training, poorly written procedures, etc., that can

influence the likelihood of a human error. PSFs are discussed and addressed in practically all human reliability analysis (HRA) approaches adopted in PRA. For this reason, and because we are not explicitly concerned here with probabilities, PSFs of the kinds commonly addressed in HRA will not be explicitly modeled in the case studies that we present in this report and we will primarily concentrate our modeling effort on the representation of plant conditions as error-forcing contexts for HFEs. However, PSFs that are considered characteristic of and specifically associated with team-related processes, such as communication and coordination, will be explicitly considered as possible contributors to error-forcing contexts.

5.2 Development of Screening Guidelines

In this section, we present some guidelines for the screening of sequences and situations for which a DFM analysis would be applied. Because DFM analysis requires the commitment of resources, it is useful to develop and make available for use in future analyses a set of screening guidelines for the selection of those scenarios, among all that may theoretically be considered within a nuclear plant PRA, for which the additional modeling effort required by the execution of a DFM team effect analysis appears to be justified.

5.2.1 Review of NUREG/CR-6093 and NUREG/CR-6208

NUREG/CR-6093 (Barriere, et. al., 1994), applies to Low Power and Shutdown (LP&S) operations, and NUREG/CR-6208 applies to Full Power operations.

NUREG/CR-6093 identifies human error of commission events as primarily initiating events in an accident sequence, although historical data (LER reports and summaries) show that some events are intermediate event(s) that inhibit or degrade the operation of safety systems. NUREG-6093 finds that errors of commission are a significant fraction of the errors in the LER database that was the subject of the study. This result applies to both PWR and BWR power reactors. LP&S accident sequences are not included in IPE PRAs. However errors of commission (EOC) are included in the Full Power IPEs, but not usually as initiating events. The predominance of errors of commission over errors of omission appears to characterize LP&S operations in a much more marked way than full power operations.

Unlike Full Power operations, all classes of human actions and errors (i.e., initiator, pre-accident, and recovery) seem to play a significant role in LP&S operations and events. Human induced initiators, both inside and outside the control room, comprise a significant portion of observed errors. In addition, there are frequently dependencies between the activities leading to an initiating event and those required for a most expeditious recovery response. In particular, human-initiated events usually are not explicitly treated in full power PRAs. The more direct human-system interactions that are characteristic of LP&S operations can result in mistakes which, in turn, lead to errors of commission. In contrast, the human errors that are explicitly modeled in full power PRAs are typically errors of omission (for example, the NRC Generic Letter 88-20 does not require errors of commission to be modeled in licensee Individual Plant Examinations), and when mistakes are included, only errors in the control room are typically modeled. The DFM model lends itself to the modeling of team related errors of commission that are accident initiators as well as to the modeling of the dependencies among the conditions (states) that lead to such events. These dependent human actions impact the progression of LP&S events. Such dependencies include, for example, temporal phase-crossings (e.g. initiator and post-accident dependencies) and separate erroneous actions by several groups caused by incorrect labeling of equipment.

Large numbers of multiple concurrent tasks are possible during LP&S conditions, whereas multiple tasks are not as common under full power conditions. Examples of important multiple concurrent tasks include simultaneous performance of different surveillance tests which create multiple RCS draindown paths and maintenance activities which result in plant conditions (e.g., increasing sump or tank volumes) identical to those caused by control room errors, thereby hindering diagnosis of LP&S events.

Other findings of NUREG/CR-6093 are that procedures are frequently deficient, either by providing inadequate guidance or in omitting instructions for unexpected contingencies while performing evaluations. This is especially troublesome with temporary procedures for special evolutions during shutdown. Procedures are important in modeling human errors in full-power PRAs; however, for LP&S operations that are performed under complicating conditions, much greater emphasis is placed on manual control actions. Also, personnel not normally at the plant (e.g., headquarters engineers and contractors), who are not intimately familiar with the plant's day-to-day work

practices and normal operating procedures may be performing tasks that can affect safety. Operators face continuously changing plant conditions and configurations. Communications within task teams is very important and much of the equipment is operated manually, and response to LP&S events are often achieved through manual actions rather than automatic equipment response.

NUREG/CR-6208 (Roth, et. al, 1994) discusses simulated Full Power Operations and examines operator performance in cognitively demanding simulated emergencies. Highly prescriptive emergency operating procedures (EOPs) are followed by the operating crews but higher-level cognitive activities are required to carry out the (EOPs) under specific conditions. The study examined crew performance variants of two cognitively demanding simulated emergencies: (1) an Interfacing System Loss of Coolant Accident (ISLOCA) and (2) a Loss of Heat Sink (LHS) scenario complicated by a leaking pressurizer power operated relief valve (PORV). Data were collected using training simulators at two plant sites. Up to 11 crews from each plant, including both actual operator crews currently on shift and staff crews, participated in each of two simulated emergencies for a total of 38 cases analyzed. A number of situations were found where situation assessment and response planning enabled the crews to handle aspects of the situation that were not fully covered by the procedures. These included:

- An EOP step that explicitly requested that crews identify and isolate a leak on their own;
- A case where the procedure containing relevant guidance could not be reached within the EOP transition network;
- Cases where operators needed to determine whether plant behavior was the result of known manual or automatic actions (e.g.), a controlled cooldown) or the result of a plant fault;
- Cases where the appropriateness of the procedure steps required evaluation and in some cases required redirection of the procedure path;
- A case where operators had to decide whether to manually initiate a safety system based on consideration and balancing of multiple goals related to safety.

The results also clarified the role of group interaction in situation assessment and response evaluation, and provided suggestive evidence of the conditions under which crew interaction skills may be expected to affect technical performance of crews. There were significant differences among the teams relative to identifying the correct plant condition in the beginning of the sequence, as well as the plant condition as a result of their actions as time progresses. Some teams were not able to transition to the correct EOP from an incorrect EOP. The Behaviorally Anchored Rating Scales (BARS) developed by Montgomery et al. (1992), that are presented in Appendix D of the report, may be useful in assisting the definition of variable states in the DFM models.

5.2.2 Safety-Important Operator Actions Identified by Westinghouse User Group

With respect to full-power operations, the Westinghouse Users Group has identified a set of safety related activities that are believed to be important from the standpoint of being heavily conditioned by operator actions. These activities, which are explicitly considered and included by most utilities in their PRAs, are:

- Establish Cold Leg recirculation - Large LOCA - CCW to RHR,
- Establish Cold Leg recirculation - Small LOCA - CCW to RHR
- Manually start Auxiliary Feedwater - transients
- Isolate ruptured Steam Generator - SGTR
- Initiate feed-and-bleed
- Terminate safety injection - SGTR
- Manually actuate reactor-trip - transients
- Maintain turbine-driven Auxiliary Feedwater - SBO
- Terminate safety injection - secondary side breaks
- Establish Emergency Boration - ATWS

5.2.3 Screening Guidelines

The report and study findings summarized in the preceding Sections 5.2.1 and 5.2.2 can be translated, using the ATHEANA methodology framework, into a set of screening guidelines to modify and augment the traditional PRA development process. Although there are differences between the importance of the factors and conditions that lead to human team errors of commission between LP&S and full-power operations, these differences do not impact the general criteria that may be used to screen scenarios and select sequences as appropriate candidates for analysis of team effects. Thus, the approach taken was to develop the guidelines to encompass both types of operations, rather than having a separate set of guidelines for each of the two types of operations. The screening guidelines consider both errors of commission and errors of omission for both accident initiating events and intermediate safety system related events in an accident sequence.

The screening process begins by examining the dominant accident sequences identified in a PRA followed by less dominant sequences that have the potential to be risk significant if team human errors can result in an additional initiating event(s) or intermediate (progression) event(s) related to safety system functions. One must also consider totally new accident sequences not yet represented in a PRA, particularly those that may result from errors of commission that result in an accident initiating event, followed by a progression of events that bears little similarity to events that are represented in the PRA. The identification of such accident sequences is perhaps the most demanding part of the screening process and requires a detailed review of the normal operating procedures, the EOPs and the power reactor technical specifications.

In order to ensure coverage in a PRA of the types of safety related system functions that the screening process should apply to, a Team Human Error/Unsafe Action Master Logic Diagram (MLD) was developed and is shown in Figure 5.1. This MLD is based on the sources of information that we have discussed in Sections 5.2.1 and 5.2.2, i.e., NUREG/CR-6093 (Barriere, et. al., 1994), NUREG/CR-6208 (Roth, et. al, 1994), and information from the Westinghouse User's Group, which applies to PWRs in general. Each of the functions in the MLD can be examined in accordance with the generic event sequence diagrams (ESDs) shown in Figures 5.2 and 5.3 to develop a generic accident event sequence. To convert the generic sequence to a power reactor specific sequence it is necessary to review the specific procedures, appropriate instrument readings, procedural steps, and particulars of potentially unsafe and safe shutdown conditions.

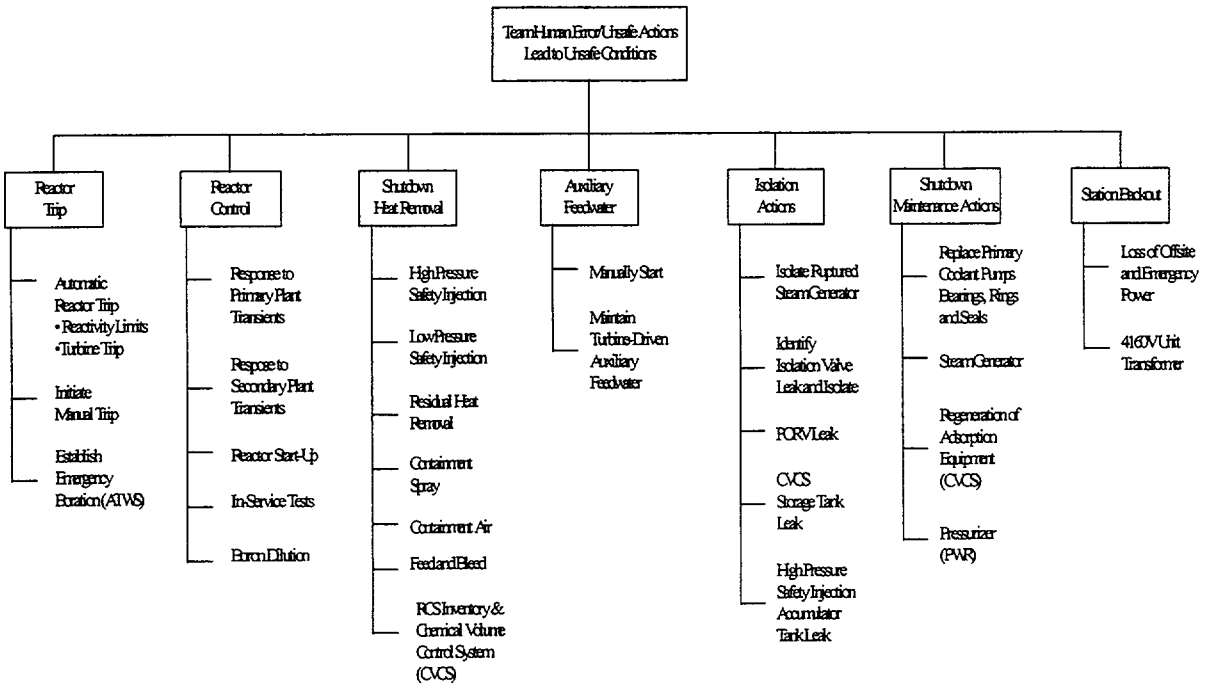


Figure 5.1: Team Human Error/Unsafe Action Master Logic Diagram (MLD)

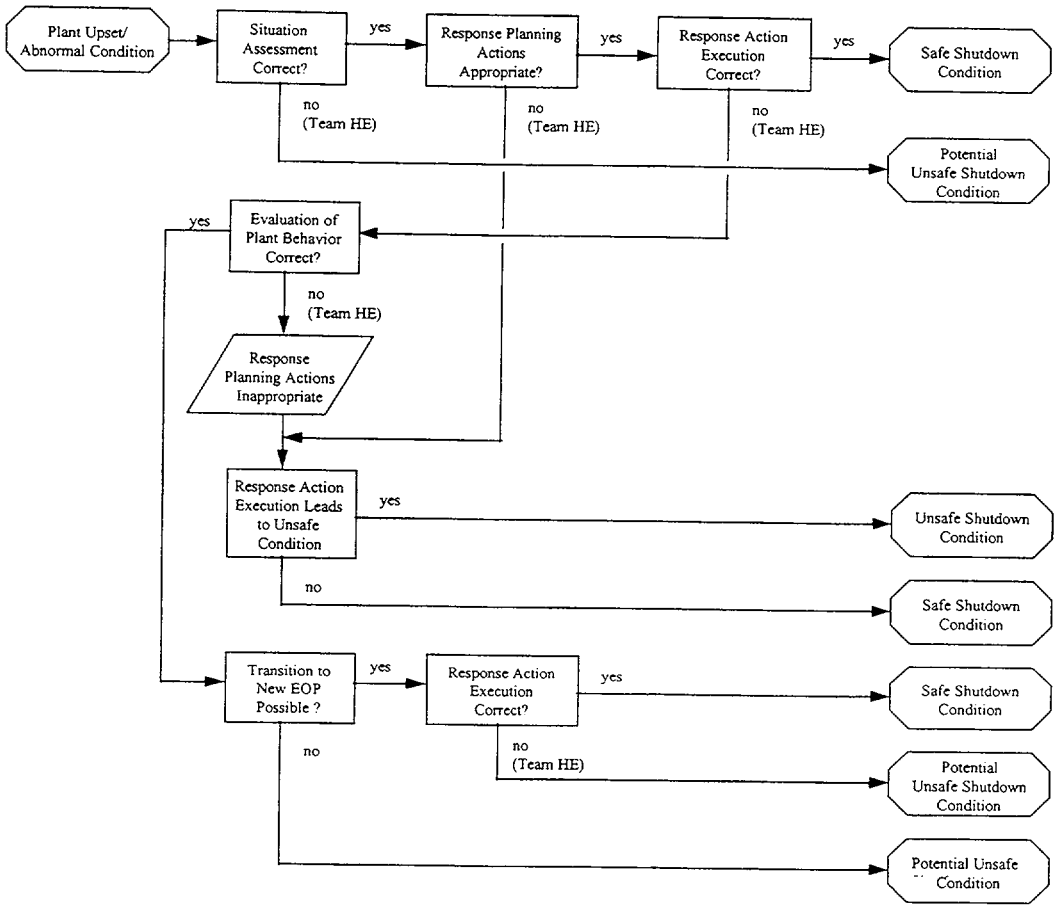


Figure 5.2: Team generic ESD - situation assessment/response planning/response execution

The result of the screening process is a set of accident sequences that include potentially important contributions from team effects and that should be compared to the dominant sequences in the existing plant PRA to decide whether they should be seriously considered for detailed modeling and analysis and eventual inclusion in the PRA. The screening guidelines correspond to the steps of the screening process and can be summarized as follows:

1. Review the Team Human Error/Unsafe Action MLD for applicability and make modifications as appropriate for a specific plant.
2. For each MLD safety related function, cross-reference to the generic ESDs and develop generic accident sequences.
3. Convert generic accident sequences to plant-specific accident sequences after a review of appropriate system drawings and specifications as well as team procedures.
4. Compare with dominant accident sequences identified in the PRA.

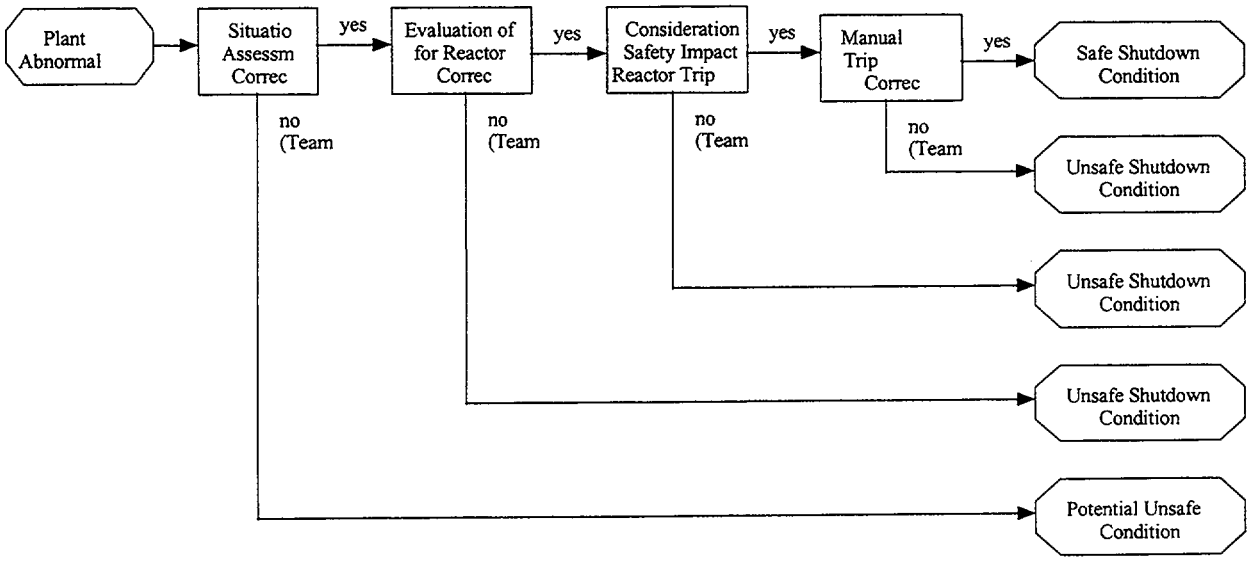


Figure 5.3: Team generic reactor trip ESD

5.3 Identification of Unsafe Actions and Error Forcing Contexts

At the beginning of the process, a human failure event is given for which unsafe actions and associated error-forcing contexts are to be identified. In general, an HFE will be directly related to a particular safety system or function in the context of a specific accident scenario (an initiating event and one of the safety systems in the associated event tree in PRA terms). The goal is to identify which combinations of unsafe actions and error forcing contexts can lead to the safety system or function becoming disabled or inoperable, leading to a degraded unsafe plant condition.

The framework for DFM-based team effects analysis consists of four basic, separate processes: a screening process (which we have discussed above in Section 5.2.3), an information gathering process, a modeling process and an analysis process.

The information gathering process can be summarized as follows:

1. Identify safety systems associated with HFE.
2. Identify the procedures that pertain to the accident sequence in question and the step(s) in the procedures relevant to the HFE.
3. Identify what information is required by the operators in order to follow the procedures and monitor the effectiveness of their actions.
4. Identify instruments that provide the information, either directly or indirectly.
5. Identify how the instruments can provide misleading information.

With respect to the modeling process, the DFM model to be constructed consists of four main parts: the model of the plant, the model of the instruments, the model of the cognitive behavior of each member of the team and the model of the interaction between the team members.

The analysis of the model consists of three stages and two types of analysis. The two types of analysis are forward simulation and back-tracing to find prime implicants. The first stage of the analysis consists of using forward simulation to verify that the model makes sense. The second stage consists of using back-tracing, with the HFE as the top event, to identify prime implicants. The resulting prime implicants will contain the unsafe actions and the

associated error-forcing contexts. The third stage consists of using forward simulation with the prime implicants of interest as boundary conditions to understand how the unsafe actions and error forcing contexts lead to the HFE.

The salient modeling features of the DFM framework are illustrated and discussed in detail in the following Section 5.4, which also presents some examples of analytical results obtained for a "development case study." A more complete discussion of the analysis results obtainable by application of the framework is found in Chapter 6, which presents and discusses a "demonstration case study."

5.4 DFM Modeling Structure

This section presents the dynamic flowgraph structure that is used to model human performance and team effects. The first part of this section discusses some structural elements for specific error forcing contexts and resulting human errors. The second part presents a development case study model that was used to aid in the development of the structure. This case study is presented here to illustrate the key concepts of the DFM modeling structure.

5.4.1 DFM Treatment of Error Forcing Contexts

This section describes how the DFM modeling structure for such issues that are related to human performance and team effects.

5.4.1.1 Unavailable or Misleading Instrumentation

In the DFM modeling structure, the possibility of unavailable or misleading indications due to unavailable or malfunctioning instrumentation can be explicitly considered. To illustrate how the DFM modeling structure treats this issue, consider the simple model module shown in Figure 5.4.

In Figure 5.4, node P represents the actual state of a physical process parameter, where the state is a discrete range of continuous values, or a discrete system or component state. Node PI represents the observable indication of the variable represented P. Node PI would typically have the same states as node P, with the possible addition of a 'no indication' state to account for the indicator being completely unavailable. Node IS represents the status of the indicator. Typically node IS would have a state that represents the normal behavior of the indicator, and one or more 'faulted' states. In the 'normal' state, the states of node P map to the corresponding states of node PI. If node IS is in an 'unavailable' state, then all states of node P map to the 'no indication' state of node PI. A variety of other faulted states, each with an associated 'fault model', are possible. In the 'generic' fault model, each state of node P maps to all states of node PI (except possibly the actual state of node P). In other words, in the generic fault model, if the indicator is faulted, its reading can be any state of the parameter (except possibly the actual state of the parameter). In the 'stuck-at-X' fault model, all states of node P map to state X of node PI. In other words, the indicator always reads a value of X, regardless of the actual parameter value. Note that the stuck-at-X fault model is subset of the generic fault model.

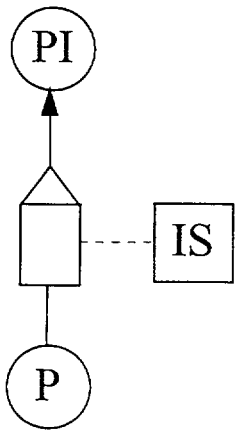


Figure 5.4: Simple model illustrating unavailable and misleading indications

5.4.1.2 Errors in Monitoring and Detection

In the DFM modeling framework the root causes and effects of errors in monitoring and detection can be explicitly considered. Root causes of errors in monitoring and detection include an operator failing to look at an instrument or an operator misreading an instrument. To illustrate how the DFM modeling framework treats this issue, consider the simple model module shown in Figure 5.5.

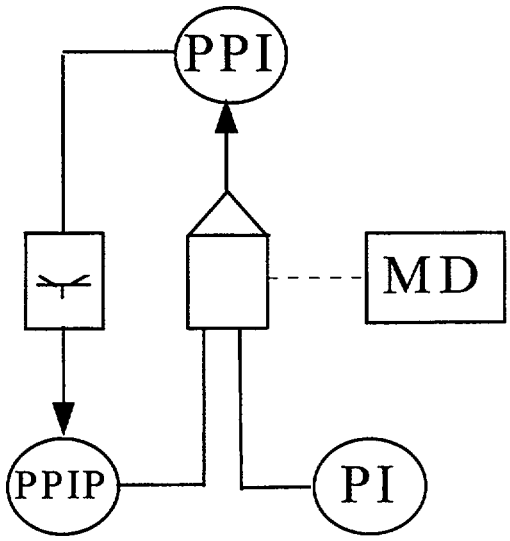


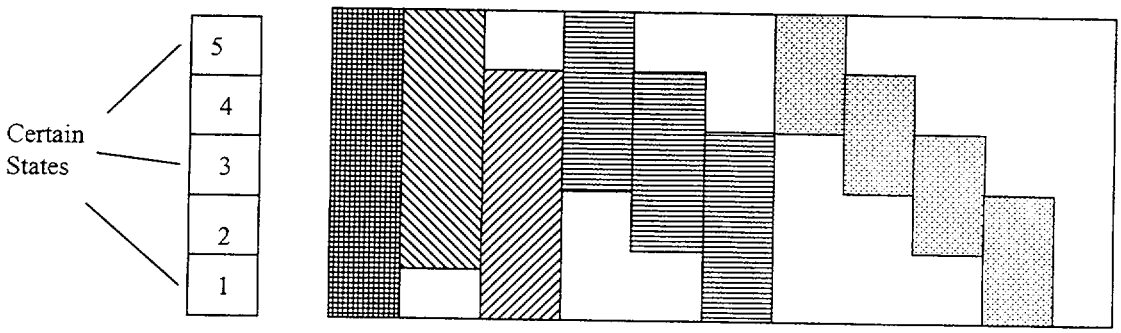
Figure 5.5: Simple model illustrating errors in monitoring and detection.

In Figure 5.5, node PI represents the observable indication of the value of a physical parameter or the state of a system or component and Node PPI represents that indication as perceived by the operator. Typically node PPI will have the same states as node PI. Node PPIP represents the operators perceived parameter indication from the previous time step. Node MD represents whether or not there is an error in the monitoring or detection process. Typically, node MD would have three states, where one state represents that there is no error, and the other two states representing two types or root causes of errors in the monitoring or detection process: 'operator fails to read indicator' and 'operator misreads indicator'. Note that the types of monitoring and detection errors that can be modeled are not limited to these two.

For the state of node MD in which there is no error in monitoring or detection, the states of node PI map directly to the corresponding states of node PPI. In other words, the value of the indication perceived by the operator is the same as the actual value of the indication. For the state of node MD in which the operator fails to read the indicator, the states of node PPIP map directly to the states of node PPI. In other words, the value of indication perceived by the operator is the same as it was in the previous time step. For the state of node MD in which the operator misreads the indicator, each state of node PI maps to any state of node PPI except the state that corresponds to the current state of node PI. In other words, the value of the indication perceived by the operator can be anything but the actual value of the indication.

5.4.1.3 Operator Assessments of Plant Parameters or System States

An issue in the modeling of monitoring or situation assessment activities is the representation of an operator's assessment of a physical plant parameter (pressure, temperature, level) or system states (availability, reliability). Typically a given plant parameter or system state variable is represented by a single node that contains the same states as the node that represents the actual information source for that parameter or state variable. In addition, the assessment node may contain some states that represent uncertainty or vagueness in the operator's assessment. Such uncertainty could arise from conflicting or missing information, or from information that conflicts with the operator's prior assessment, possibly conditioned by a factor that affects how the operator updates his assessment.



4+3+2+1 = 10 possible uncertain states

Figure 5.6: Possible uncertain states for a node with 5 certain states

The uncertain states for a given assessment node could range from a representation using one state to represent complete uncertainty over the range of certain states, to a representation with (N-1) uncertainty states, where each uncertainty state represents that the operator is unsure if the actual value is one of two adjacent certain states. All intermediate representations in between these two are also possible, as illustrated in Figure V.6. Note that this assumes that uncertain states are modeled as uniform distributions. Other truncated distributions can be used that limits the number of possible states. However, it is difficult to justify non-uniform distributions in a deterministic model.

The level of detail appropriate for the modeling of uncertain states is one of the issues that was investigated during this research. It is discussed in association with specific examples of the DFM structure.

5.4.1.4 Updating of Operator Assessments

As discussed in Section 4.3.1.1, the operators' mental model consists of their understanding of the current plant state and behavior, which relates to their expectations of future plant states and behaviors as the accident progresses. The mental model considers multiple parameters and time sequence considerations. The operators update their mental model based upon information received about the plant state, both in terms of system states and process conditions. The operators use their mental model to make a situation assessment and to evaluate the effect of their actions on the plant condition. In this subsection, a structure for modeling the formulation and evolution of a mental model for plant parameters is presented.

Figure 5.7 shows a simple DFM model for the updating of an operator's assessment of one parameter based on the information provided by one indicator. Node PIP represents the operator's perception of the reading provided by the indicator and node PP represents the operator's prior assessment of what the value of the parameter should be at this time. Node ISAP represents the operator's prior assessment of the status of the indicator (i.e., whether it is working properly or not). Node PA represents the operator's current assessment of the parameter, based on the states of the three above mentioned nodes and node ISA represents the operator's current assessment of the status of the indicator. Node CF represents cognitive factors that could affect how the operator updates his assessment of the parameter and the indication. The assessments of the parameter and the indicator are coupled, in that the states of the two must be consistent in any given time step, e.g., the operator cannot use the reading of the indicator to update his parameter assessment if he believes that the indicator is wrong. Note that the above framework does not specify a specific model for the cognitive updating of situation assessment.

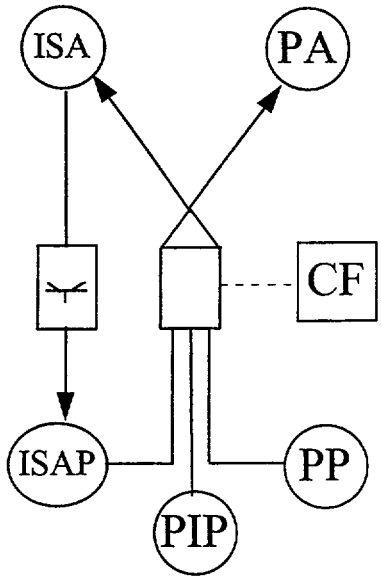


Figure 5.7: Simple model illustrating operator assessment of a plant parameter

5.4.1.5 Operator's Mental Model of Plant Behavior

In the DFM modeling framework the operator's mental model of plant behavior can be explicitly considered. More specifically, an operator's mental model of how plant parameters change over time can be modeled. For example, consider the simple DFM model shown in Figure 5.8.

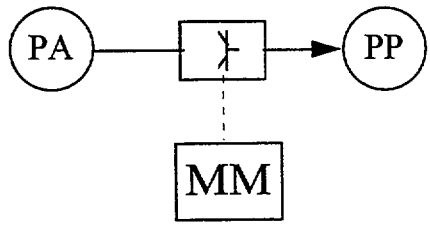


Figure 5.8: Simple model for operator's mental model of plant behavior

In Figure 5.8, node PA represents the operator's current assessment of a parameter and node PP represents the operator's prediction of what it should be in the next time step. Node MM represents the operators mental model of how the parameter changes over time, in the situation that could be defined by other input nodes into the transfer box. The states of node MM would typically indicate whether or not the operator's mental model is correct, and if it is incorrect, whether it is too fast, too slow, etc.

5.4.1.6 Operator Diagnosis State

The operators' diagnosis state can also be modeled at a level of detail that that is commensurate with the discussion in Section 5.4.1.3 above. The diagnosis state is expressed as a high level description of the plant state, usually related to which procedures the operators should be following, i.e., loss of coolant accident (LOCA).

5.4.1.7 Modeling of the individual control room operators and the communication between them.

In the DFM modeling structure, the individual operators in the control room team can be considered in terms of he communication between them and their individual responses base on their individual situation assessments. An example of how the communications aspect would be handled in a DFM model is illustrated in the simple model shown in Figure 5.9. Node P represents a variable to be communicated from one operator to another, and node CP represents the value that is actually communicated. Node CF represents factors that can affect the communication process. The states of node CF can represent the various root causes of communication failures, as well as a state that represents successful communication.

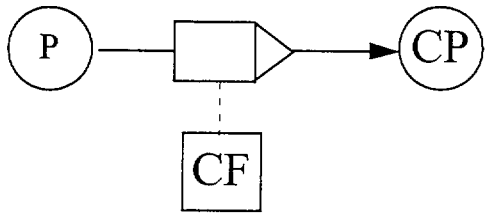


Figure 5.9: Simple DFM model for communication between operators

5.4.1.8 Modeling of Response Planning Activities

Response planning is the process of deciding what actions to take. The operators' situation assessment of the plant state, and the available procedures are used to formulate a response plan. One type of unsafe action related to the response planning activity is a circumvention, which can happen if the operators decide that a step in the procedures is inappropriate or unnecessary based on their assessment of the situation. Another type of error related to response planning that can occur is if the operators implement the wrong procedures based on their understanding of the situation. After a response plan has been implemented, the operators will evaluate the effectiveness of their actions through monitoring activities.

Root causes of errors in response planning, if the crew is following written procedures, can include reading the procedures wrong or intentionally skipping or disregarding the procedures. The latter can be broken down further. If the crew is not following written procedures, then root causes of errors in response planning would primarily be due to incomplete or incorrect knowledge. How root causes of errors in response planning are modeled in DFM is generally problem dependent.

5.4.2 Development Case Study

This section presents the case study used to aid in the development of the overall modeling methodology, and to identify modules for inclusion in the module library. This scenario was used as an interim test case, so that we could verify our methodology as it was being developed. The development case study is the same as the test case presented in the Phase I report, however it is extended to consider additional issues associated with human cognitive behavior and team effects.

5.4.2.1 Overview of Development Case Study Scenario

The scenario modeled in the development case study is a loss of shutdown cooling during PWR mode 5 operations, in which the primary coolant temperature is less than 212°F and is drained to a level between the middle and the top of the Hot Leg (mid-loop). During mid-loop operations, shutdown cooling is provided by using the low pressure safety injection system (LPSI) pumps to pump primary coolant from the bottom of a hot leg, through a shutdown cooling heat exchanger, and then back to the core via injection through one or more of the Cold Legs. For the scenario of interest, the control room team consists of a Senior Reactor Operator (SRO), a Reactor Operator (RO), and an Auxiliary Reactor Operator (ARO). Personnel outside the control room may include various maintenance groups, plant operators, and roving patrols.

Upon initiation of Loss of Shutdown Cooling, an alarm inside the control room alerts the control room operators, and the Loss of Shutdown Cooling Abnormal Operating Procedures (AOPs) are initiated. Upon receipt of the alarm(s), the control room supervisor initiates containment closure by first making an announcement to evacuate personnel inside the containment and then telling the Containment Work Manager to close the personnel and equipment hatches. The CRS will then assign one of his operators to monitor the most vital plant parameters (Reactor Coolant System (RCS) level, Shutdown Cooling System (SDCS) flow and core exit temperature). If the RCS level is less than 21 inches in the hot leg or is lowering, the operators are instructed to initiate procedures to recover RCS inventory. If the RCS level is adequate and stable, but the SDC flow is less than 2200 gpm or the LPSI Pump amperage is not normal, then the operators are instructed to initiate procedures to recover SDC flow. If the RCS level and the SDC flow are adequate, but the Core exit temperature is greater than 200 °F or is rising, then the operators are instructed to initiate procedures to recover the Reactor Core exit temperature.

At this point, if the cause of the loss of shutdown cooling has been diagnosed as an RCS leak, the SRO proceeds to the steps to recover the RCS Inventory, where the procedures ask for the cessation of all existing perturbations, which include, but are not limited to, various types of draining evolutions and primary sampling. The operators inside the control room check the selected flow path to make sure the coolant does not bypass the core through known leaks, and then starts an operable or available High Pressure Safety Injection (HPSI) Pump. Two Cold Leg Injection Valves or one Hot Leg Injection are throttled to establish flow, depending on the selected flow path. If the cause of inventory loss is unknown or has occurred through a cold leg breach, then flow should be directed through Hot Leg Injection. If there are no HPSI pumps available, then the operators are instructed to use an available Containment Spray pump, which involves throttling open the Discharge Valve. The proper alignment of the Containment Spray system requires a remote operator to perform actions outside the control room.

Once RCS injection has been initiated, the operators are instructed to fill the RCS to a final level at least 5 inches higher than the initial level and higher than or equal to 30 inches in the Hot Leg. If this cannot be accomplished, then they are instructed to provide alternate Reactor Core cooling.

The development case study explicitly considered the operator activities associated with the diagnosis of the cause of loss of shutdown cooling and the subsequent RCS inventory recovery actions. This involves the monitoring and assessment of several parameters by different operators in the team, and communication of the value of parameters to the control room supervisor. It may also involve communication with personnel outside the control room. This scenario thus involves all of the cognitive activities discussed in Chapter 4: detection and monitoring, situation assessment, response planning and response implementation.

5.4.2.2 Overview of Phase 1 Test Case Operator Actions

In sub-step a) of Step 3 of the Loss of Shutdown Cooling AOPs, if the RCS level is less than 21 inches in the hot leg or if the RCS level is lowering, then a loss of RCS inventory is indicated, and the operators are instructed to go to Step 4 to recover RCS inventory. In sub-step b, if the SDC flow is less than or equal to 2200 gpm or if the pump amperage is abnormal, then a loss of SDC flow is indicated, and the operators are instructed to go to Step 5 to recover SDC flow. In sub-step c, if the core exit temperature is greater than 200°F or if it is rising, then the operators are instructed to go to Step 6 to recover core exit temperature. Thus, the operator, in this case the SRO, must monitor the critical parameters, and, based on their values, assess the situation, and decide on which actions to carry out next, i. e., plan a response.

The Development Case Study only considered the monitoring of the RCS level. The Reactor Operator (RO) monitors the level, and when he determines that it is either below 21 inches in the Hot Leg or is lowering, he notifies the SRO of that fact. The SRO then decides on the appropriate course of action and instructs the RO to carry out that action.

In AOP step 4 a), which is the sub-step that contains the key operator actions for the recovery of RCS inventory, the RO is instructed to use the HPSI system for the injection of emergency coolant. If the HPSI system is unavailable, the RO is instructed to use the Containment Spray system, which requires a remote operator (in the pump room). There is a note instructing the operator to use Hot Leg injection rather than Cold Leg injection if there is a Cold Leg break or if the cause of inventory loss is unknown. The RO must, thus, assess the situation (the status of the key systems) and plan and implement a response.

In the Development Case Study, the control room team must diagnose the cause of the loss of shutdown cooling and take appropriate action. In the development case study, only the case in which a leak has caused a loss of RCS inventory is considered, and thus, only whether or not the operators correctly initiate recovery of RCS inventory is modeled. Therefore, the first important decision to be made by the control room team is whether or not to initiate the recover RCS inventory procedures (Step 4 of the AOP). This decision involves the monitoring and assessment of two key parameters: the RCS level and the RCS level change, both of which are measured by the same instrumentation.

Once the operators have correctly assessed the situation and have initiated the proper procedures, they must decide which system to use for emergency coolant injection (HPSI or Containment Spray), and where to inject it into the RCS (the Hot Leg or Cold Leg).

5.4.2.3 Description of Development Case Study DFM Model

The purpose of the model for the development case study, and the analysis that is applied to it, is to identify unsafe actions and associated error forcing contexts that could lead to a human failure event. For the loss of shutdown cooling scenario, a human failure event could be the failure to use the Containment spray for emergency coolant injection in the case in which there is a primary system leak and the HPSI is unavailable. During the midloop stage, if emergency cooling is not provided, core uncovery could result in as little as 20 minutes. Figure 5.10 shows the DFM model for the analysis of the HFE mentioned above for the loss of shutdown cooling scenario. The variables and node abbreviations are shown in Table 5.I. The model representation can be summarized by noting that nodes RC, RL, RLS, HPS, CSS, HPF, CSF and L represent the plant parameters, whereas nodes RLI, RCI and IG represent variables associated with monitoring and detection activities, nodes RCP, RLP , RCA, RLA and AS1 and associated connections represent variables associated with situation assessment activities, nodes D1, RP1, D2 and RP2 represent variables associated with response planning activities, nodes RI1 and RI2 represent variables associated with response implementation activities, and nodes CP1, CD1, CP2, CD2, CP3 and CD3 represent variables associated with team communication activities. The reader can find full detail and explanation of the development case study model in Appendix A.

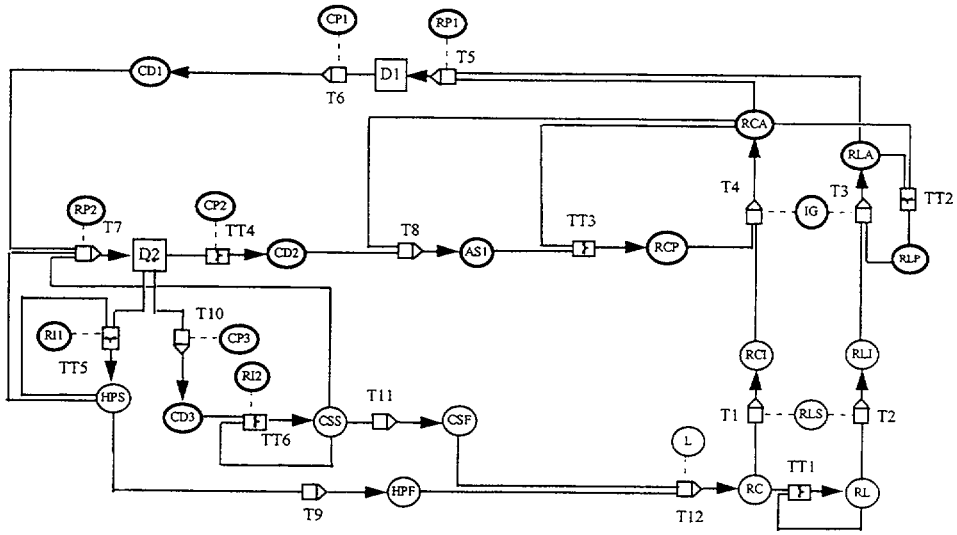


Figure 5.10: DFM Model for loss of shutdown cooling scenario

Table 5.I: Variables and Node Abbreviations Used in Figure 5.10

Node	Abbreviation
RCS Level Change	RC
RCS Level	RL
RCS Level Change Indication	RCI
RCS Level Indication	RLI
RCS Level Sensor Status	RLS
Root Cause of Error in Information Gathering Process	IG
SRO's Assessment of RCS Level Change	RCA
SRO's Assessment of RCS Level	RLA
SRO's Predicted RCS Level Change	RCP
SRO's Predicted RCS Level	RLP
SRO's Decision of Whether to Initiate Recovery of RCS Inventory	D1
Root Causes of Errors in Response Planning	RP1, RP2
Decision Communicated to RO	CD1
Root Causes of Communication Errors	CP1, CP3
RO's Decision of Whether to use HPSI or Containment Spray	D2
SRO's Assessment of Activity Status	AS1
Root Causes of Errors in Feedback Communications	CP2
Feedback Communication from RO to SRO	CD2
HPSI System Status	HPS
Root Causes of Errors in Response Implementation	RI1, RI2
HPSI Flow	HPF
Operator 2's Communication to Remote Operator	CD3
Containment Spray System Status	CSS
Containment Spray Flow	CSF
RCS Leak	L

5.4.2.4 Node RC

Node RC represents the rate of change of the RCS level. This node has three states, shown in Table 5.II below. For the purposes of this model, the operators are only concerned with whether the RCS level is increasing, decreasing or stable. Thus the continuous parameter of the rate of change of the RCS level is discretized into the three states shown in Table 5.II.

Table 5.II: States for Node RC

State	Description
I	RCS level is increasing
S	RCS level is stable
D	RCS level is decreasing

5.4.2.5 Node RL

Node RL represents the coolant level in the RCS. In this example, three states for the RCS level is an absolute minimum. There must be a state for an RCS level of greater than 21 inches in the Hot Leg, in which case there is adequate core cooling and the operators are not instructed to recover the RCS inventory (unless the RCS level is decreasing), i. e., the core cooling safety function is not threatened. There must also be a state for an RCS level of less than 21 inches in the Hot Leg but greater than the level of the top of the fuel, in which case the operators are instructed to recover RCS inventory, but there is still adequate core cooling, i. e., the core cooling safety function is

adequate but threatened. The remaining state is the RCS level for which the core is uncovered. This last state is the undesirable state that would be used for a top event. Thus the states of Node RL are shown in Table 5.III.

Table 5.III: States for Node RL

State	Description
A	adequate - greater than 21" in the hot leg
B	barely adequate - less than 21" in the hot leg but above the top of the active fuel
I	inadequate - below the top of the active fuel

5.4.2.6 Node RLS

Node RLS represents the state of the RCS level sensor. Although there are three systems for RCS level monitoring, the RWLP system, the HJTC system and the RWLI system, we assume for simplicity in this example that there is only one sensor. It is straightforward to model three sensors, although that may result in very large decision tables. Since node RLS represents a hardware state, it is, by nature, a discrete node. However, in addition to the normal reliable state of the sensor, in which the value of the RCS level indicated is the actual value, we must postulate one or more failed states. One obvious state is the case in which the sensor is completely broken and does not provide any reading at all (or a reading of zero). For the time being we will consider only these two states (shown in Table 5.IV), although in the future we may add more failed states.

Table 5.IV: States for Node RLS

State	Description
N	normal
F	failed

5.4.2.7 Node RCI

Node RCI represents the indication of the rate of change of the RCS level. Node RCI has the same states as node with the addition of the no reading state, as shown in Table 5.V.

Table 5.V: States for Node RCI

State	Description
I	increasing
D	decreasing
S	stable
N	no reading

5.4.2.8 Node RLI

Node RLI represents the indication of the RCS level. The states for node RLI are the same as those for node RL with the addition of the no reading state, as shown in Figure VI.

Table 5.VI: States for Node RLI

State	Description
A	adequate
B	barely adequate
I	inadequate
N	no reading

5.4.2.9 Node IG

Node IG represents the information gathering process of the operator monitoring the RCS level. This node is used to model errors in information gathering. The possible errors considered here include: the operator misreads the RCS level, the operator misinterprets the RCS level change and the operator does not read the sensor. The States for node IG are shown in Table 5.VII.

Table 5.VII: States for Node IG

State	Description
R	operator reads instrument correctly
N	operator does not read instrument
H	operator misreads RCS level high
L	operator misreads RCS level low
S	operator perceives the RCS level as stable when it is changing
I	operator perceives the RCS level as increasing when it is stable
D	operator perceives the RCS level as decreasing when it is stable

5.4.2.10 Development Case Study Analysis

In this section, back tracing analysis on the demonstration case study model is used to identify combinations of Error Forcing Contexts (EFCs) and Unsafe Actions (UAs) that can lead to an Human Failure Event (HFE). In this example the HFE is the failure of the control room team to initiate recovery of RCS inventory procedures before the RCS level reaches an inadequate state.

In the development case study model, each time step is approximately 1 minute long. This is based on how long it would take the Senior Reactor Operator) SRO to notice an abnormal condition, decide what to do about it, communicate to the Reactor Operator (RO) and receive feedback. With this size of a time step, it would take many to traverse the 'barely adequate' state of the RCS level. Many time steps were traversed by the model and it was found that only a small number of steps, in this state, are necessary to get meaningful prime implicants. Thus, the backtracking analysis will go back three time steps.

The top event, in terms of node states at each of the four steps is shown in Table 5.VIII. The RCS level is assumed to start in the 'adequate' state at time step -3, make a transition to the 'barely adequate' state for time steps -2 and -1, and then end up in the 'inadequate' state at time step 0. The states of three other nodes are specified as an 'initial condition' for time step -3. The operators are assumed to be in a normal responsive state; thus, node RCP is in the 'stable' state for time step -3, node RLP is in the 'adequate' state for time step -3 and node CD2 is in the 'not confirmed' state for time step -3.

Table 5.VIII: Top event states

Node		Condition		Time
Name	Description	State	Description	
RL	RCS level	2	adequate	-3
RL	RCS level	1	barely adequate	-2
RL	RCS level	1	barely adequate	-1
RL	RCS level	0	inadequate	0
RLP	predicted RCS level	2	adequate	-3
RCP	predicted RCS level change	1	stable	-3
CD2	confirmation from RO to SRO	1	not confirmed	-3

A few consistency rules are also specified to ensure meaningful prime implicants are generated. First, the 'unavailable' state of nodes HPS and CSS are defined to be 'sink states', i.e. they do not recover in future time steps. Second, nodes L and RLS are defined to be constant; they must always be in the same state, whatever that may be.

The 78 prime implicants that result from the backtracing analysis are given in Appendix B. The prime implicants can be divided into 4 groups, with respect to hardware states:

1. Contains RLS = 1 ('failed') at time step -3.
2. Contains HPS = 1 ('available') at time step -3,
3. Contains HPS = 2 (unavailable') and CSS = 1 ('available) at time step -3.
1. Contains HPS = 2 (unavailable') and CSS = 2 ('unavailable) at time step -3.

In the prime implicants of the first group, the 'failed' state of node RLS means that the SRO never knows that the RCS level is decreasing, and thus he never decides to do anything about it. Within the second group, in which the HPSI system is available, the following combinations of operator states result in the top event:

A. Either RI1 = 1 ('slip'), CP1 = 1 ('communication error'), IG = 1 ('does not read indication') or RP1 = 1 ('disregards') at time step -3 and either RI1 = 1 ('slip') or CP1 = 1 ('communication error') at time step -2.
B. Either RI1 = 1 ('slip'), CP1 = 1 ('communication error') or RP1 = 1 ('disregards') at time step -3 and RP1 = 1 ('disregards') and IG = 1 ('does not read indication') at time step -2.
C. IG = 1 ('does not read indication') at time step -3 and IG = 1 ('does not read indication') at time step -2.

Within the third group, in which the HPSI system is unavailable but the Containment Spray system is available, the following combinations of operator states result in the top event:

D. Either RI2 = 1 ('slip'), CP3 = 1 ('communication error'), CP1 = 1 ('communication error'), IG = 1 ('does not read indication') or RP1 = 1 ('disregards') at time step -3 and either RI2 = 1 ('slip'), CP3 = 1 ('communication error') or CP1 = 1 ('communication error') at time step -2.
E. Either RI2 = 1 ('slip'), CP3 = 1 ('communication error'), CP1 = 1 ('communication error') or RP1 = 1 ('disregards') at time step -3 and RP1 = 1 ('disregards') and IG = 1 ('does not read indication') at time step -2.
F. IG = 1 ('does not read indication') at time step -3 and IG = 1 ('does not read indication') at time step -2.

In both the second and third groups, all of the prime implicants contain an error state in the first and second time steps. This can be generalized to say that an error must occur in every time step that the model spends in a state in which the RCS level is barely adequate and decreasing. In other words, if the operators make an error, they have many chances to recognize the error and correct it. Two other observations can be made about the second and third groups of prime implicants. First, RP = 1 (SRO disregards procedures) cannot be the single error after the first time step; it must be combined with IG = 1 (SRO does not read indication or reads the wrong instrument).

6. SELECTION AND ANALYSIS OF CASE STUDY

6.1 ISLOCA Scenario Background

After using a "development case study" as discussed in Ch. 5, an Interfacing System Loss of Coolant Accident (ISLOCA) at full power was chosen as a 2nd case study to demonstrate the scenario screening procedure and to validate the overall methodology. The ISLOCA scenario involves a leak from the high pressure Reactor Coolant System (RCS) into the low pressure Residual Heat Removal (RHR) System. The description of this scenario is taken from NUREG/CR - 6208 (Roth, et. al., 1994). Two isolation valves between the hot leg of the RCS system and the RHR system that are normally kept closed and de-energized begin to leak, which produces an increase in pressure in the RHR, resulting in a break in the RHR piping in the Auxiliary Building approximately five minutes into the event.

The first alarms that come in are an RHR discharge high pressure alarm and pressurizer pressure and level low alarms, which results in a reactor trip approximately 30 seconds later. At this point the crew is instructed to turn to the Reactor Trip and SI Procedure (E-0) in the EOPs. They reach a step in the procedure that asks if the RCS is intact. By that point the Pressurizer Relief Tank (PRT) has ruptured, resulting in radiation inside the containment. Therefore the answer is no, and the EOPs direct a transition to the Loss of Reactor or Secondary Coolant Procedure (E-1). There is a step later in the E-0 procedure that checks for Auxiliary Building radiation symptoms and if the answer is yes, directs them to an ISLOCA procedure. However, the operators would transition to E-1 before they get to that step unless they notice a problem in the RHR and diagnose a LOCA into the RHR system early on, and make a transition to the ISLOCA procedure without explicit procedural guidance. Once in E-1 there is no explicit transition to the ISLOCA procedure.

The ISLOCA scenario is difficult from the situation assessment point of view. A situation can arise in which the operating crew has to identify and isolate the leak into the RHR without explicit procedural guidance. Although the Emergency Operating Procedures (EOPs) contain procedures for identifying and isolating an ISLOCA, it is possible for a situation to arise in which the operating crew cannot reach the ISLOCA procedure within the EOP network, because plant symptoms generated early in the event are similar to the pattern of symptoms that are produced by a Loss of Coolant Accident (LOCA) inside containment. Thus, if the timing of events is right, the EOPs will direct the operators to a procedure for a LOCA inside containment. Once in the LOCA procedure, there is no explicit transposition to the ISLOCA procedure. Early detection of a problem in the RHR is important because it provides the potential for isolating the leak into the RHR before the RHR piping bursts.

6.2 Demonstration of Methodology

6.2.1 Screening Process

An ESD for this scenario is shown in Figure 6.1. A team initiating event was chosen, "Two RHR/RCS Isolation Valves Left Open". This event can be caused by a lack of proper communication between the Reactor Control and Auxiliary Building crews, or a error of commission by the Auxiliary Building Crew.

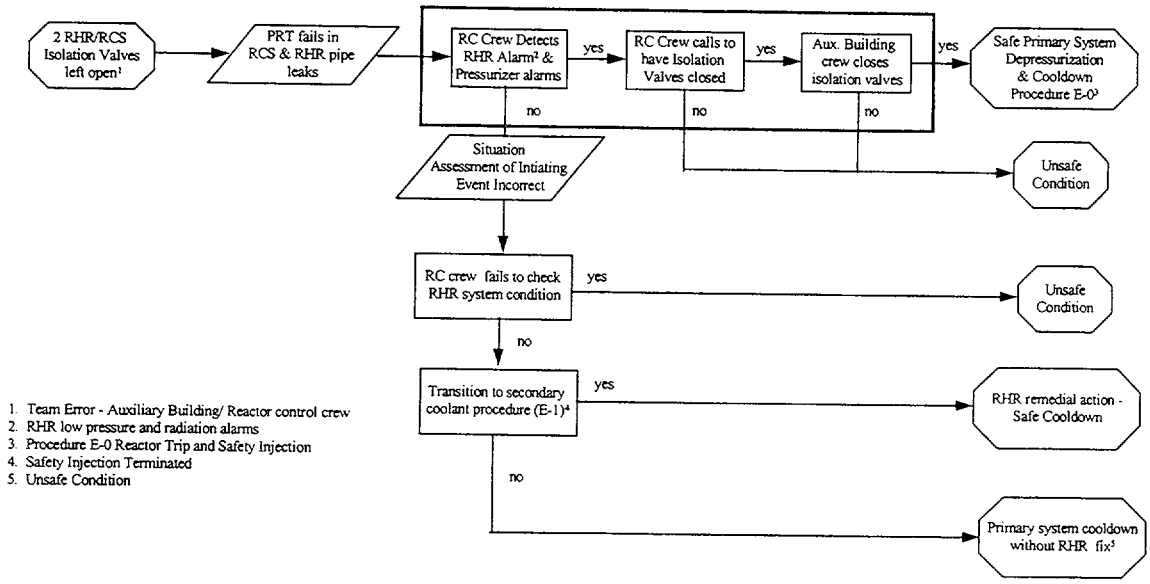


Figure 6.1: ISLOCA event sequence diagram

This class of accidents (ISLOCA) has been shown in Independent Plant Examination (IPE) reports, e.g. San Onofre, Units 2 and 3, to account for approximately 3% of the total core damage frequency (CDF). Therefore including this sequence for further consideration of team errors of commission seems justified. The ESD is constructed in such a way that human failure events can be identified - both errors of omission and errors of commission.

6.2.2 Identification of Human Failure Events (HFEs)

Human failure events represent the failure of a function, system, or component as a result of an unsafe action or sequence of actions that results in a worsened plant condition. A human failure event can be either an error of commission (EOC), in which the operators intentionally disable or terminate a necessary safety function or intentionally initiate an inappropriate system, or an error of omission, in which the operators inadvertently fail to initiate a required safety system or function. Note that an HFE implies failure to recognize and recover from the error in time to prevent the plant's transition to a degraded state.

Given this definition of an HFE, and the ESD shown in Figure 6.1, two possible HFEs associated with the ISLOCA scenario can be identified: a) someone intentionally opens the RCS/RHR isolation valves (error of commission) and b) the operating crew fails to isolate the leak before the RHR piping bursts (error of omission).

6.2.3 Identification of Unsafe Actions and Error Forcing Contexts

To validate the methodology, unsafe actions and error forcing contexts for the human failure event "operating crew fails to isolate the leak before the RHR piping bursts" were identified. If the crew does isolate the leak before the RHR piping burst, they prevent any release of radioactivity from the containment. Note that event trees are generally not developed for the ISLOCA initiating event, because it is assumed that the leak cannot be isolated and therefore the event leads directly to core damage. However, in our postulated initiating event in which the isolation valves are inadvertently opened, the leak can be isolated.

6.2.4 DFM Model

Figure 6.2 shows the system portion of the DFM Model for this example. The following sections describe the nodes and decision tables for the model.

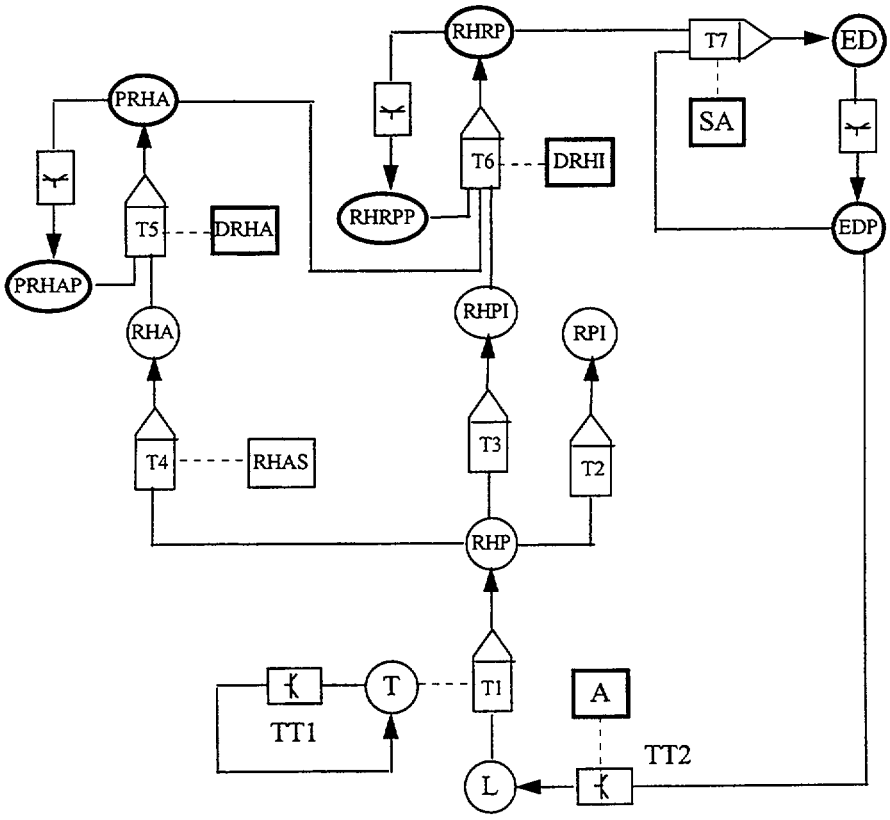


Figure 6.2: DFM model for ISLOCA example

6.2.4.1 Node T

Node T represents the current time. In the description of the scenario presented above, it takes 5 minutes from when the symptoms of a problem in the RHR appear to the bursting of the RHR piping. One time step in the DFM model is the time it would take the crew to notice a problem in the RHR, diagnose a leak from the RCS into the RHR, make a transition to the ISLOCA procedure and isolate the leak. In this example, we have assumed three time steps will elapse before the RHR piping bursts. Therefore node T has four states, shown in Table 6.I below.

Table 6.I: States of node T

State	Description
-3	time step -3
-2	time step -2
-1	time step -1
0	time step 0

6.2.4.2 Transition Box TT1

Transition box TT1 controls the progression of time in the model; specifically, it increments the state of node T for each time transition. The decision table for transition box TT1 is shown in Table 6.II. There is no row for state 0 because this transition never occurs in the model.

Table 6.II Decision table for transition box TT1

Node T		Node T	
State	Description	State	Description
-3	time step -3	-2	time step -2
-2	time step -2	-1	time step -1
-1	time step -1	0	time step -1

6.2.4.3 Node L

Node L represents whether or not there is leakage into the RHR from the RCS. The states of node L are shown in Table 6.III.

Table 6.III: States of node L

State	Description
L	leak
N	no leak

6.2.4.4 Node RHP

Node RHP represents the RHR system pressure. The states of node RHP are shown in Table 6.IV. The 'very high' state is defined to be the pressure at which the RHR piping will burst. The 'high' state is defined to be a pressure for which measurements would indicate a problem with the RHR system.

Table 6.IV: States of node RHP

State	Description
N	Normal
H	High
V	Very High

6.2.4.5 Transfer Box T1

Transfer box T1 models the dependence of the RHR pressure on the presence of a leak and the time step. The decision table for transfer box T1 is shown in Table 6.V. The RHR pressure makes a transition to the 'very high' state in the third time step if there is still a leak form the RCS.

Table 6.V: Decision table for transfer box T1

Node L		Node T		Node RHP	
State	Description	State	Description	State	Description
-	-	-3	time step -3	H	high
-	-	-2	time step -2	H	high
-	-	-1	time step -1	H	high
L	leak	0	time step 0	V	very high
N	no leak	0	time step 0	H	high

6.2.4.6 Node RPI

Node RPI represents whether or not the RHR piping has burst, and has the two states shown in Table 6.VI.

Table 6.VI: States of node RPI

State	Description
I	intact
B	burst

6.2.4.7 Transfer Box T2

Transfer box T2 models the integrity of the RHR piping on the RHR pressure. The decision table for transfer box T2 is shown in Table 6.VII. The piping bursts if the RHR pressure is in the 'very high' state

Table 6.VII: Decision table for transfer box T2

Node RHP		Node RPI	
State	Description	State	Description
N	normal	I	intact
H	high	I	intact
V	very high	B	burst

6.2.4.8 Node RHPI

Node RHPI represents the indications of the RHR pressure on the control board. The states for node RHPI are shown in Table 6.VIII.

Table 6.VIII: States of node RHPI

State	Description
N	high
H	normal

6.2.4.9 Transfer Box T3

Transfer box T3 maps the RHR pressure to the RHR discharge pressure indications on the control board. Because the indicators are highly redundant, it was judged that a failure of all of them is not credible (Julius, et. al., 1995). The decision table for transfer box T3 is shown in Table 6.IX.

Table 6.IX: Decision table for transfer box T3

Node RHP		Node RHPI	
State	Description	State	Description
N	normal	N	not high
H	high	H	high
V	very high	H	high

6.2.4.10 Node RHA

Node RHA represents whether or not the RHR high pressure alarm has come on. The states of node RHR are shown in Table 6.X.

Table 6.X: States of node RHA

State	Description
A	alarm
N	no alarm

6.2.4.11 Node RHAS

Node RHAS represents the status of the RHR high discharge pressure alarm. Since the RHR High Pressure Alarm is not highly redundant, its failure is modeled. The states for node RHRA are shown in Table 6.XI.

Table 6XI: States of node RHAS

State	Description
N	normal
M	malfunctioning

6.2.4.12 Transfer Box T4

Transfer box T4 models the dependence of the actuation of the RHR high discharge pressure Alarm on the actual RHR pressure and the status of the alarm. The decision table for transfer box T4 is shown in Table 6.XII.

Table 6.XII: Decision table for transfer box T4

Node RHP		Node RHAS		Node RHA	
State	Description	State	Description	State	Description
N	normal	-	-	N	no alarm
H	high	N	normal	A	alarm
H	high	M	malfunctioning	N	no alarm
V	very high	N	normal	A	alarm
V	very high	M	malfunctioning	N	no alarm

6.2.4.13 Nodes PRHA and PRHAP

Node PRHA represents whether or not the control room crew has detected the RHR high discharge pressure alarm. If the crew detects the alarm early enough and realizes that there is a problem with the RHR, they may be able to isolate the leak into the RHR before the piping bursts. This node is one of the crews' situation assessment state nodes.

Node PRHAP carries the state of node PRHA over to the next time step. This node is a modeling artifice that ensures that the control room crew does not make a transition from a state in which the alarm has been detected to a state in which the alarm has not been detected, consistent with the concept that the crew cannot make a transition to a more general state of situation assessment (Gertman, et. al., 1996). The states for nodes PRHA and PRHAP are shown in Table 6.XIII.

Table 6.XIII: States of nodes PRHA and PRHAP

State	Description
A	alarm detected
N	alarm not detected

6.2.4.14 Node DRHA

Node DRHA represents whether or not the control room crew detects the RHR high discharge pressure alarm during the current time step. The states for node DRHA are shown in Table 6.XIV. The 'irrelevant' state is used when the alarm has not gone off; in this case the crew cannot detect the alarm (or fail to detect the alarm). Also, if the crew has already detected the alarm, the first two states of node DRHA again have no meaning.

Table 6XIV: States of node DRHA

State	Description
D	detects alarm
N	fails to detect alarm
X	irrelevant

6.2.4.15 Transfer Box T5

Transfer box T5 models the dependence of the control room crews' state of RHR high discharge pressure alarm detection on the actual presence of the alarm and whether or not the alarm has already been detected. The decision table for transition box T5 is shown in Table 6.XV.

Table 6.XV: Decision table for transfer box T5

Node RHA		Node PRHAP		Node DRHA		Node PRHA	
State	Description	State	Description	State	Description	State	Description
A	alarm	A	alarm detected	X	irrelevant	A	alarm detected
A	alarm	N	alarm not detected	D	detects alarm	A	alarm detected
A	alarm	N	alarm not detected	N	fails to detect alarm	N	alarm not detected
N	leak	N	alarm not detected	X	irrelevant	N	alarm not detected

6.2.4.16 Nodes RHRP and RHRPP

Node RHRP models whether or not the control room crew realizes there is a problem with the RHR system. If the crew realizes that there is a problem with the RHR early enough, they may be able to isolate the leak into the RHR before the piping bursts. This node is one of the crews' situation assessment state nodes.

Node RHRPP carries the state of node RHRP over to the next time step. This node is a modeling artifice that ensures that the control room crew does not make a transition from a state in which they think there is a problem with the RHR to a state in which they do not think there is a problem with the RHR. The states for nodes RHRP and RHRPP are shown in Table 6.XVI.

Table 6.XVI: States of nodes RHRP and RHRPP

State	Description
P	RHR problem
N	no RHR problem

6.2.4.17 Node DRHI

Node DRHI models whether or not the control room crew detects the RHR symptoms on the control board (RHR discharge pressure and RHR discharge temperature). The states for node DRHI are shown in Table 6.XVII. This node is meaningful only if the crew has not already detected the RHR symptoms.

Table 6.XVII: States of node DRHI

State	Description
D	detects RHR
N	does not detect RHR
X	irrelevant

6.2.4.18 Transfer Box T6

Transfer box T6 models the dependence of the control room crews' state of detection of a problem with the RHR system on whether or not they have detected the RHR high discharge pressure alarm and the readings of the RHR indicators on the control board. The decision table for transition box T6 is shown in Table 6.XVIII. It is assumed if the crew detects the alarm they will always look at the control board and detect the RHR symptoms. This is consistent with what is reported in NUREG/CR-6208 (Roth et. al., 1994), where all five crews who detected the RHR high discharge pressure alarm looked at the control board for confirmation. Failure of the crew to look at the control board is possible but is less likely and is not reflected in the decision table.

Table 6.XVIII: Decision table for transfer box T6

Node RHPI		Node PRHA		Node RHRPP		Node DRHI		Node RHRP	
State	Description	State	Description	State	Description	State	Description	State	Description
N	normal	N	alarm not detected	N	no RHR problem	X	irrelevant	N	no RHR problem
H	high	A	alarm detected	N	no RHR problem	-	-	P	RHR problem
H	high	N	alarm not detected	N	no RHR problem	D	detects RHR	P	RHR problem
H	high	N	alarm not detected	N	no RHR problem	N	does not detect RHR	N	no RHR problem
H	high	-	-	P	RHR problem	X	X	P	RHR problem

6.2.4.19 Nodes ED and EDP

Node ED represents whether or not the control room crew has diagnosed the ISLOCA and has made the transition to the ISLOCA procedures (given that they have detected a problem with the RHR). The states of node ED are shown in Table 6.XIX. Node EDP carries the state of node ED over to the next time step. This node is a modeling artifice that ensures that the control room crew does not make a transition from a state in which they are in the ISLOCA procedures to a state in which they never went to the ISLOCA procedures

Table 6.XIX: States of nodes ED and EDP

State	Description
D	ISLOCA diagnosed
N	ISLOCA not diagnosed

6.2.4.20 Node SA1

Node SA1 models whether or not the control room crew diagnoses the ISLOCA and makes the transition to the ISLOCA procedures during the current time step (given that the have not done so already). The states for node SA1 are shown in Table 6.XX. This node is meaningful only if the crew has not already diagnosed the ISLOCA.

Table 6.XX: States of node SA1

State	Description
E	makes diagnosis
N	does not make diagnosis
X	irrelevant

6.2.4.21 Transfer Box T7

Transfer box T7 models the dependence of the control room crews' state of diagnosis of the ISLOCA on whether or not they have detected a problem with the RHR system. The decision table for transition box T7 is shown in Table VI.XXI.

Table 6XXI: Decision table for transfer box T7

Node RHRP		Node EDP		Node SA1		Node ED	
State	Description	State	Description	State	Description	State	Description
P	RHR problem	D	ISLOCA diagnosed	X	irrelevant	D	ISLOCA diagnosed
P	RHR problem	N	ISLOCA not diagnosed	D	diagnoses ISLOCA	D	ISLOCA diagnosed
P	RHR problem	N	ISLOCA not diagnosed	N	does not diagnose ISLOCA	N	ISLOCA not diagnosed
N	no RHR problem	N	ISLOCA not diagnosed	X	irrelevant	N	ISLOCA not diagnosed

6.2.4.22 Node A

Node A represents whether or not the control room crew makes a slip while attempting to isolate the leak into the RHR system. The states of node A are shown in Table 6.XXII. This node is meaningful only if the crew has diagnosed the ISLOCA.

Table 6.XXII: States of node A

State	Description
S	slip
N	no slip
X	irrelevant

6.2.4.23 Transition Box TT2

Transition box TT2 models the dependence of the leak into the RHR on whether or not the control room crew has successfully isolated it. The decision table for transition box TT2 is shown in Table 6.XXIII.

Table 6.XXIII: Decision table for transition box TT2

Node ED		Node A		Node L	
State	Description	State	Description	State	Description
E	ISLOCA diagnosed	S	slip	L	leak
E	ISLOCA diagnosed	N	no slip	N	no leak
N	ISLOCA not diagnosed	X	irrelevant	L	leak

6.2.5 DFM Analysis

In this section the DFM analysis is described and the results are presented.

6.2.5.1 Top Event and Consistency Rules

The top event for the deductive analysis should correspond to the human failure event for which the unsafe actions and error forcing contexts are to be identified; in this case, the control room crew fails to isolate the leak before the RHR piping bursts. The top event for this analysis is shown in Table 6.XXIV. The first condition in the top event (RPI = B) indicates that the RHR piping has burst by the final time step. The second condition (T = 0) is used to control the timing of the scenario, mainly that the RHR piping will burst after three time steps if the leak is not isolated. The last three conditions are initial conditions. They specify that the control room crew is in an initial state in which they have not yet detected the RHR high discharge pressure alarm, they have not yet detected a problem with the RHR system and they have not yet diagnosed the ISLOCA.

Table 6.XXIV: Top Event

Node	State	Time
RPI	B (burst)	0
T	0 (time step 0)	0
PRHAP	N (alarm not detected)	-3
PRHPP	N (no RHR problem)	-3
EDP	N (ISLOCA not diagnosed)	-3

One consistency rule is also specified for the analysis; namely that node RHAS is constant. In other words, the alarm is either normal for all time steps in the analysis, or it is malfunctioning for all time steps in the analysis. Intermittent malfunctioning of alarms is possible and could be reflected in the model by increasing the number of states.

6.2.5.2 Prime Implicants

The deductive analysis resulted in 25 prime implicants, which are exhaustively listed in Appendix C. Each prime implicant can be interpreted as a sequence of unsafe actions that leads to the top event HFE. Two of the prime implicants are shown below:

(RHAS = 'Normal', DRHA = 'does not detect alarm' and DRHI = 'does not detect RHR problem') at time = -3, DRHA = 'does not detect alarm' and DRHI = 'does not detect RHR problem') at time = -2 and DRHA = 'detects alarm', SA1 = 'diagnoses ISLOCA' and A1 = 'slip') at time = -1.

(RHAS = 'malfunctioning' and DRHI = 'does not detect RHR problem') at time = -3, (RHAS = 'malfunctioning' and DRHI = 'does not detect RHR problem') at time = -2 and (RHAS = 'malfunctioning, DRHI = 'detects RHR problem', SA1 = 'diagnoses ISLOCA' and A1 = 'slip') at time = -1.

In the first prime implicant three unsafe actions (crew fails to detect RHR high discharge pressure alarm, crew fails to detect RHR problem, and crew makes a slip attempting to isolate the leak) combine to cause the top event HFE (two of the unsafe actions in combination are stretched over two time steps). In the second prime implicant, there are two unsafe actions (crew fails to detect RHR problem, and crew makes a slip attempting to isolate the leak). The first unsafe action occurs in combination with an error forcing context (RHR high discharge pressure alarm is malfunctioning) to combine to cause the top event HFE.

7. EXTENSIONS OF DFM SOFTWARE

7.1 Development of DFM Modules

In this section the development of a set of pre-made DFM sub-modules for commonly encountered situations is described. The library enables the user of the DFM Model Editor software to select an entire DFM module from the library and include it in a model, if the situation is appropriate. Table 7.I provides a list of the modules included in the Module Library.

Table 7.I: List of DFM modules in Module Library

Module Name	Description
PI	Parameter Indication Module
PIX	Extended Parameter Indication Module
C	Communication Module
CF	Communication Module (with feedback)
SA1	Parameter Assessment Module (one indication)
SA2	Parameter Assessment Module (two indications)
RP	Response Planning Module
RE	Response Execution Module

As an example of a DFM module, consider the module for a communication-acknowledgment loop between two individuals, shown in Figure 7.1. Node D represents a response planning choice or decision, taken or made by the first individual. Node CD represents the choice or decision actually communicated to the second individual and node CP1 represents root causes for failures in that communication process. Node A represents the feedback or acknowledgment communicated back to the first individual from the second individual, and node CP2 represents root causes of failure in that communication process.

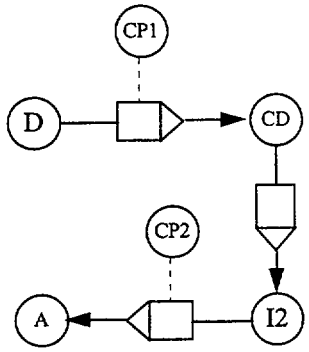


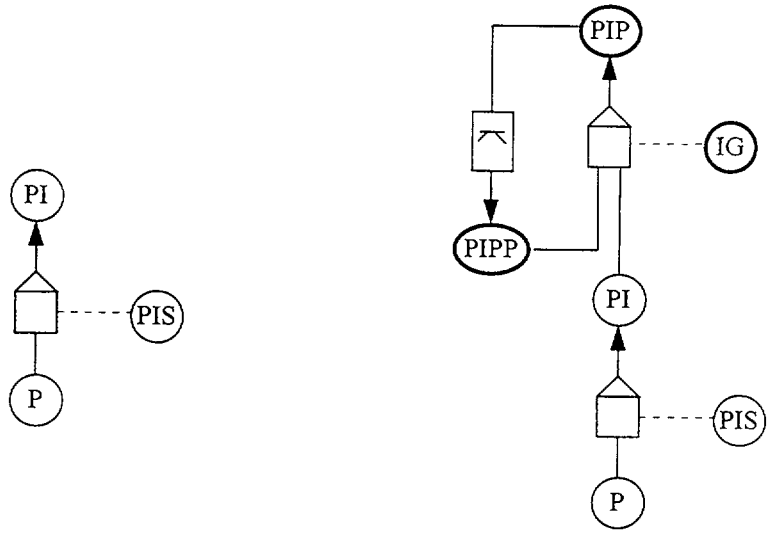
Figure 7.1: DFM module for communication-acknowledgment loop between two individuals

In the DFM model Editor, the module shown in Figure 7.1 could be selected by the user and placed in a DFM model whenever a situation is encountered that involves a communication-acknowledgment loop between two individuals. The user would be able to connect other nodes and transfer boxes to the nodes in the module, and modify the properties of the objects from the default values provided with the module.

7.1.1 Parameter Indication Module

Figure 7.2 shows two versions of the Parameter Indication (PI) module, which models the frequently encountered situation of a physical or system related parameter and associated indication. The second version, shown in Figure 7.2b, also includes the operators perception of the indication.

The basic version of the PI module, shown in Figure 7.2a, consists of a node that represents the parameter in question (node P in Figure 7.2), a node that represents the indication of the parameter (node PI in Figure 7.2) and a node that represents the status of the indicator (node PIS in Figure 7.2). The extended version of the PI module, shown in Figure 7.2b, also includes two nodes that represents an operator's perception of the parameter indication (nodes PIP and PIPP in Figure 7.2b) and a node that represents root causes of errors in the cognitive activity of perceiving the parameter indication (node IG in Figure 7.2a). Node PIPP is a dummy node that maps the states of node PIP to the next time step.



a) PI Module

b) PIX Module

Figure 7.2: Parameter Indication Module

With respect to the DFM Model Editor software and the implementation of the Module Library, the user needs to specify the parameter states (states of node P) and the fault model for the indicator (states of node PIS). As a default, the indication (node PI) has states that would be identical to the parameter states (node P); however, the user has the ability to override the discretizations of node PI. The default fault model is the generic fault (a single faulted state) in which, if the fault exists, the indication can be any state other than the actual parameter state. Then, with the default fault model, if the user specifies the states of node P, the decision table can be automatically generated. This corresponds to the framework for modeling instrumentation described in Section 5.4.1.1.

In the extended version of the PI module (the PIX module), as a default, the node that represents the operator's perceived indication (node PIP) has the same states as the indication itself (node PI). Again, the user has the ability to override this default configuration. Node PIPP always has the same states as node PIP and the transition box is just a direct map of the states across a time step. The default configuration of the extended module corresponds to the framework for modeling monitoring and detection described in Section 5.4.1.2.

The default states for node IG are shown in Table 7.II below. If IG is in state R, in which there is no error, then the default transfer function maps the states of node PI directly to the states of node PIP. If the operator does not read the instrument (state N of node IG) then the default transfer function maps the states of node PIPP directly to the states of node PIP. In other words, if the operator misreads the indicator, then the default transfer function maps the state of node PI to any other state of node PIP. Thus, once the states of node PI are specified, the decision table for the default fault IG model can be automatically generated.

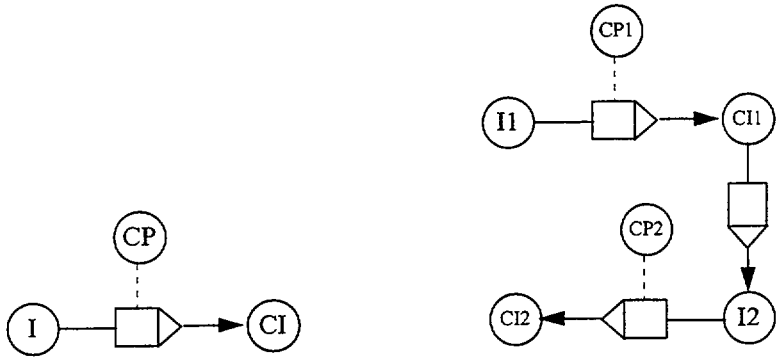
Table 7.II: Default states for node IG

State	Description
R	operator reads instrument correctly
N	operator does not read instrument
M	operator misreads instrument

7.1.2 Communication Module

Figure 7.3 shows two versions of a DFM module that can be used to represent communication between two individuals: the basic version (C Module) and the version with feedback (CF Module). The default configuration of this module is based on the framework for modeling communication between individuals, presented in Section 5.4.1.6.

Both versions of the Communication module include a node that represents the information to be communicated from the first individual to the second individual (node I in Figure 7.3a and node I1 in Figure 7.3b), a node that represents the information that is actually received by the second individual (node CI in Figure 7.3a and CI1 in Figure 7.3b) and a node that represents root causes of errors in the communication process (node CP in Figure 7.3a and CP1 in Figure 7.3b). The feedback version also includes a node that represents the information that is to be communicated back to the first individual from the second individual (node CI2 in Figure 7.3b), a node that represents the information actually communicated back to the first individual (node CI2 in Figure 7.3b) and another node that represents root causes of errors in the communication process (node CP2 in Figure 7.3b).



a) C Module b) CF Module

Figure 7.3: The Communication Module

With respect to the DFM Model Editor software and the implementation of the Module Library, the user needs to specify the states of the nodes that represents the information to be communicated (node I in Figure 7.3a and nodes I1 and I2 Figure 7.3b). In the default configuration, the nodes that represent the information actually communicated (node CI in Figure 7.3a and nodes CI1 and CI2 Figure 2b) have the same states as the previous nodes, as well as a state that indicates that no information was communicated.

In the default configuration, the nodes that represent the root causes of errors in the communication process (node CP in Figure 7.3a and nodes CP1 and CP2 in Figure 7.3b) contain the states shown in Table 7.III. One success state and four error states are modeled. Of the error states, two are sender errors and two are receiver errors.

Table 7.III: Default states of nodes CI, CI1 and CI2

State	Description
0	successful communication
1	wrong information sent
2	information is not sent
3	information sent is misunderstood by receiver
4	information is sent but not received.

In Section 4.4, eight types of sending errors and four types of receiving errors are described. The sending errors include:

1. Message content is wrong. Communication fails because the information contained in the message is incorrect.
2. Message content is inconsistent with other information. The information in the message is correct, but is inconsistent with other information available to the receiver.
3. Message content is inappropriate for the receiver. The sender fails to tailor the message for the receiver in terms of the receiver's work context, the receiver's role in the task at hand, the receiver's technical knowledge, or fails to use terminology familiar to the receiver.
4. Message production is inadequate. After an accurate and complete message has been composed and tailored to the needs of the receiver, communication can fail if the message is not produced adequately.
5. Message is not sent. In this case the sender fails to transmit a message needed by the receiver.
6. Message is sent to the wrong place or person.
7. Message is sent at the wrong time. Communication fails if the sender transmits a message either too early or too late.
8. Failure to verify message understanding. The sender fails to take positive actions to verify that the intended receiver has accepted and understood the message.

The four receiving errors include:

1. Message is not sought. If the receiver does not actively seek the information necessary to perform a task then a receiver error may occur.
2. Message is not found or not used. In this case the receiver does not find (in the case of written communication) or disregards the message.
3. Message is misunderstood.
4. Receiver does not verify message understanding. The receiver fails to take action to test his understanding of the message received.

Some of the above messages are related to cognitive activities and should be modeled as such. Sender error 2, in which the message is correct but is inconsistent with other information available to the receiver, is similar to the situation in which an operator has conflicting or inconsistent instrument readings. As such, it should not be considered a communication failure, but should be considered as a context in which the receiver can make a cognitive error in situation assessment or response planning. With respect to the receiver errors, receiver error 1, in which the message is not sought, could also be considered an error in response planning that causes the communication failure. Receiver error 2, in which the receiver disregards the message, is also actually an error in response planning, in that the root causes of the failure is operator error or misinterpretation of the message.

Of the remaining root causes, sender errors 1, 3 and 4 are grouped together as state 1 in Table 7.III, sender errors 5, 6 and 7 are grouped together as state 2 in Table 7.III, receiver errors 1 and 2 are grouped together as state 3 in Table 7.III and receiver errors 3 and 4 are grouped together as state 4 in Table 7.III. Sender error 8 only applies to feedback situations, and thus is considered to be part of state 3 for node CP 2 in Figure 7.3.

Once the information states are specified, the default decision tables can be automatically generated. For the success state of the node that represents root causes of communication errors, the transfer function is a direct map from the node that represents the information that is to be sent, to the node that represents the information that is actually received. For states 2 and 4 in Table 7.III, all states of the node that represents the information that is to be sent map to the no-information state of the node that represents the information that is actually received. For states 1 and 3 in

Table 7.III, each state of the node that represents the information that is to be sent maps to every other state of the node that represents the information that is actually received (except the no-information state).

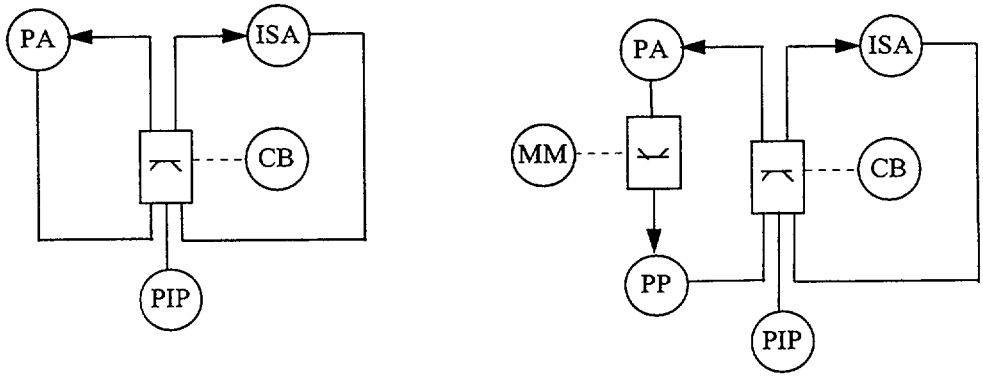
As an example, suppose node I in Figure 7.3a has two states i1 and i2, and suppose node CI has three states i1, i2 and n, where n is the 'no information received' state. In this case, the default decision table is shown in Table 7.IV below.

Table 7.IV: Example default decision table for Communication Module

Node I	Node CP	Node CI
i1	0	i1
i1	1	i2
-	2	n
i1	3	i2
-	4	n
i2	0	i2
i2	1	i1
12	3	i1

7.1.3 Situation Assessment Modules

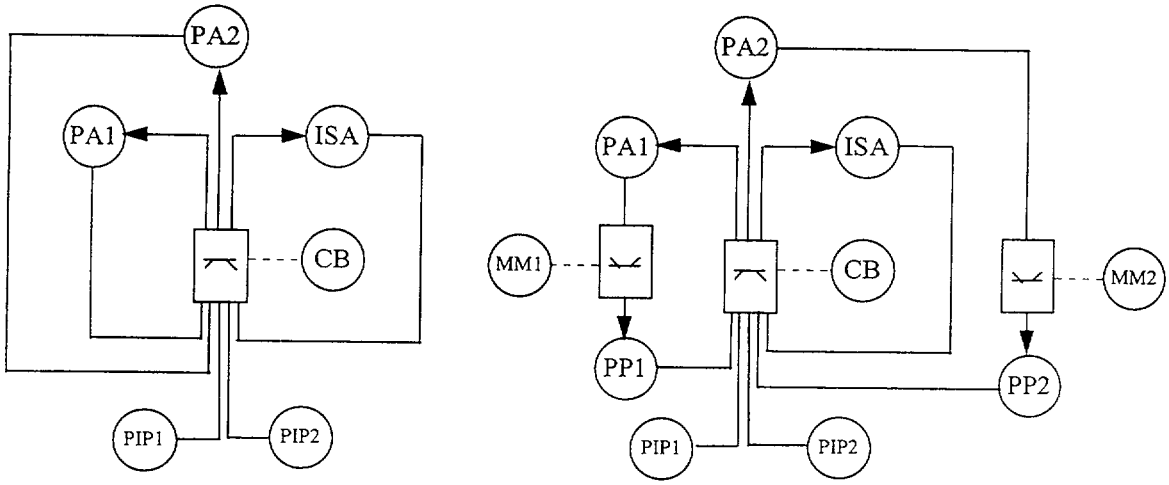
The Situation Assessment DFM modules model the operators' assessment of parameters or system states based on prior assessments. There are three types of Situation Assessment modules: the assessment of one parameter from one instrument indicator (the SA11 module), the assessment of two parameters from one indicator (the SA21 module) and the assessment of one parameter from two instrument indications (the SA12 module). Each type of situation assessment module has two versions, one in which the operator does not predict the value of the parameter in the next time step, and one in which the operator does predict the value of the parameter in the next time step (the extended or X version). This results in six modules that are shown in Figures 7.4 - 7.6.



a) SA1 module b)SA1X module

Figure 7.4: Situation Assessment module for one parameter and one instrument: a) without prediction, b) with prediction

All of the modules contain nodes that represent the operators perception of a parameter indication (node PIP in Figure 7.4 and nodes PIP1 and PIP2 in Figures 7.5 and 7.6). The SA21 and SA12 modules contain two such nodes. Note that these nodes are also part of the Parameter Indication module described in Section 7.1.1. The states of these are generally the same as the states of the nodes that represent the actual parameters (along with an 'unknown' state), as described in Section 7.1.1.

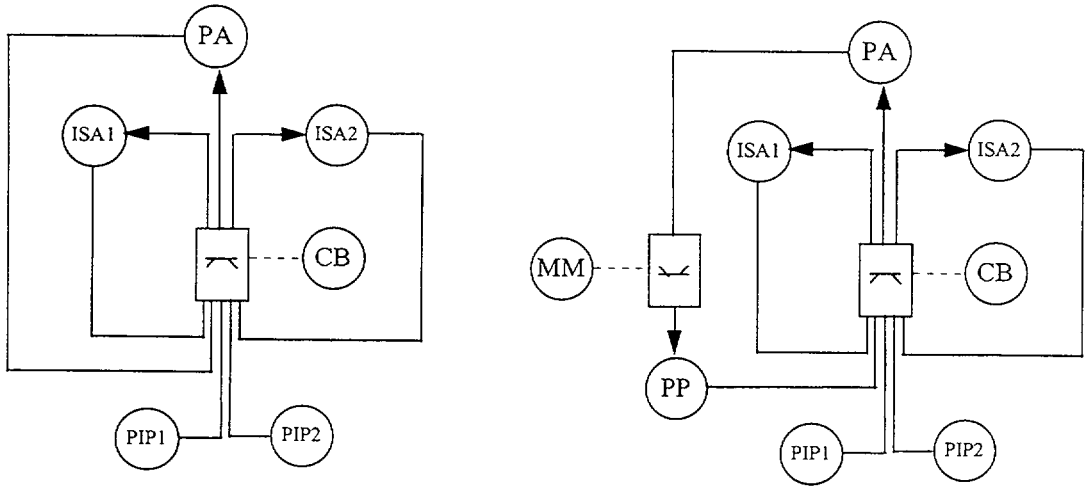


a) SA21 module

b) SA21 module

Figure 7.5: Situation Assessment module for two parameters and one indicator: a) without prediction, b) with prediction

All of the modules also contain corresponding nodes that represent the operator's assessment of the value of a parameter (node PA in Figures 7.4 and 7.6 and nodes PA1 and PA2 in Figure 7.5). The SA21 and SA12 modules contain two such nodes. These nodes also generally contain the same states as the node that represents the indicator for that parameter or state variable. In addition, the assessment node may contain some states that represent uncertainty or vagueness in the operator's assessment. Such uncertainty could arise from conflicting or missing information, or from information that conflicts with the operator's prior assessment, combined with an appropriate cognitive bias.



a) SA12 module

b) SA12X module

Figure 7.6: Situation Assessment module for one parameter and two instruments: a) without prediction, b) with prediction

The uncertain states for a given assessment node could range from one state representing complete uncertainty to (N-1) states, where each uncertain state represents that the operator is unsure if the actual value is one of two

adjacent certain states, and could include all of the states in between. The user of the module will be able to specify at which level of detail he wishes the uncertainty to be modeled.

All of the Situation Assessment modules also contain a node that represents the operator's assessment of the status of an indication (node ISA in Figures 7.4 and 7.5 and nodes ISA1 and ISA2 in Figure 7.6). The SA12 module contains two such nodes, one for each indicator. Generally, this node contains the same states as the node that represents the actual indicator status, in addition to a state that represents that the operator is uncertain of the status of the indicator.

All of the Situation Assessment modules also contain a node that represents root causes of errors in the situation assessment process (node CB in Figures 7.4 - 7.6). There is no default configuration for this node.

The extended versions of the Situation Assessment modules contain a node that represent the operator's prediction of what a parameter will be in the next time step (node PP in Figures 7.4 and 7.6 and nodes PP1 and PP2 in Figure 7.5). The SA21X module has two such nodes, one for each parameter. Generally, this node contains the exact same states as the node that represents the operator's current assessment of the same parameter. The extended versions of the Situation Assessment modules also contain a node that represents whether or not the operator's assessment of how the parameter changes with time is correct or not (node MM in Figures 7.4 and 7.6 and nodes MM1 and MM2 in Figure 7.5). The SA21X module has two such nodes, one for each parameter).

The actual form of the decision table for transition box in Figure 7.4a depends on the particular cognitive model that the user chooses; however, some general observations can be made here. If the parameter indication is consistent with the operator's prior belief and he initially believes that the indicator is reliable, there is no reason why he would not believe the indication. In addition, he would continue to believe that the indicator is reliable. On the other hand, if the operator believes that the indicator is unreliable, he will stick with his prior assessment, regardless of the indication; to do otherwise would be irrational. How the operator updates his assessment when faced with uncertain or contradictory information is a more difficult subject and requires that a mental model be developed for the specific situation assessment. Once such a model is developed, DFM can model the assessment process.

7.1.4 Response Planning Module

The Response Planning Module (RP module), shown in Figure 7.7, models an operator's decision making process. The decision is based on the operator's assessment of node PA in Figure 7.7. This is the same Node PA that appears in the Situation Assessment modules described in Section 7.1.3. As shown in that Section, two or more nodes of this type can be represented in a decision table and modeled by DFM.

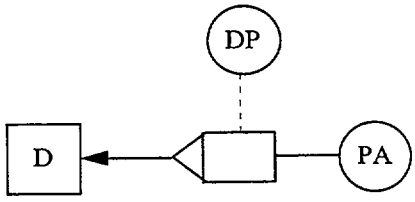


Figure 7.7: Response Planning Module

Node D represents the operator's decision. Its states represent the choices of response actions the operator can make. Node DP represents root causes of errors in the decision making (response planning) process. The default states for node DP are shown in Table 7.V.

Table 7.V: Default states for Node DP

State	Description
C	correct decision
W	wrong decision
N	no decision

The 'wrong decision' state means that the operator makes any decision other than the correct one. The actual root cause of a wrong decision is situation specific, and is thus not explicitly modeled in the RP module. The 'no decision' state indicates that the operator hesitates and fails to make a decision. Again the actual reasons for the operator's hesitation are situation specific and are not explicitly modeled in the RP module.

7.1.5 Response Implementation Module

The Response Implementation (RI) module models the implementation of the response actions decided upon by the operators in the response planning phase. The Response Implementation module is shown in Figure 7.8. Node R represents the response that has been decided upon and possibly communicated to another operator. The states of node R are typically the same as the states of the corresponding decision node. Node SS represents whether or not the response was correctly implemented, and typically has states that represent the status of the system that is the target of the response action. Node RI represents root causes of errors in response implementation. The default states of Node RI are shown in Table 7.VI. Only one error state is included; the actual error mechanisms are situation specific.

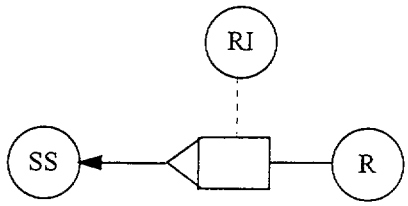


Figure 7.8: Response Implementation Module

Table 7.VI: Default states for Node RI

State	Description
S	success
E	error

8. FINDINGS, CONCLUSIONS AND RECOMMENDATIONS

The objective of this research was the development of a framework for the use of the Dynamic Flowgraph Methodology (DFM) in the area of human performance and team effects. More specifically, a framework for the identification of unsafe actions and associated error forcing contexts in specific scenarios was developed. Also developed were guidelines for screening accident scenarios for in-depth analysis using the DFM framework. Another result of the Phase II research was the identification of DFM modules for commonly encountered situations in DFM modeling of typical scenarios. These pre-built DFM modules were incorporated into a library for use in the DFM Software Toolset. The library can be built on as further applications of DFM for team related errors of omission and commission are encountered in PRA accident scenarios. The new DFM modules will be taken from the associated DFM model for a new accident scenario application.

The DFM framework developed in this project may be specifically used to complement the ATHEANA methodology. The crux of that methodology is the identification of unsafe actions and error forcing contexts, that when taken together could lead to a given human failure event. The DFM analytical approach provides an integrated modeling and analysis framework that can be applied to a specific scenario for which a human failure event has been identified. The identified combinations of unsafe actions can be assessed to determine if they are credible for that situation. Credibility is a judgementive decision that is based on the relative probability of occurrence of the unsafe actions, when compared to hardware failure events in the same accident sequence.

As part of the DFM modeling structure, guidelines for the modeling of specific elements of the human performance and team problem were formulated. Specifically, guidelines for the modeling of instrumentation, monitoring and detection, situation assessment, communication, response planning and response implementation were developed and are presented in Chapter 5 of this report. Pre-built DFM modules were developed and incorporated into the DFM Software Toolset for many of these situations.

The process for the identification of unsafe actions and error forcing contexts within specific PRA accident scenarios has been demonstrated through the use of the screening process that is presented in Chapter 5, Section 5.2, Development of Screening Guidelines. Besides demonstrating the DFM modeling and analysis procedures, the two case studies have demonstrated the integrated systems approach whereby the physical process, safety-related systems and components, and human team members are considered simultaneously and explicitly. This approach enables all three elements to be represented and analyzed with one integrated model. The advantage of analyzing a single integrated model is that hidden dependencies between the human team members and the physical system that may not be immediately obvious may be uncovered. This allows the analyst using the ATHEANA framework to include human failure events in a scenario that may otherwise not be inclusive of the appropriate consideration for human failure event contributions.

The PRA accident scenario screening guidelines and the DFM modeling structure developed in this research was shown to be a viable and effective means of identifying such scenarios and developing an integrated model. The development of a DFM model is not a trivial task. It requires a significant understanding of the problem as well as development of experience with the features of the DFM modeling technique. In general, for an analysis of depth comparable with the case studies presented in this report, the user can expect to have to go through a few iterations to reach in the modeling the necessary balance between the level of modeling detail required to give meaningful results and the need to avoid excessive detail that is likely to result in requirements for increased computational state explosion and a computationally intractable problem. At times, certain modeling subtleties and precautions are required to do the job. As a case in point, in Chapter 5 of this report a framework for modeling operator uncertainty in situation assessment was presented. The framework is in principle flexible enough to allow any level of detail that the analyst desires. In practice, the number of states can be increased beyond 10 to reflect more detail, but the added benefit of the increased level of detail should be weighed against the increased computational requirements. The two demonstration models presented herein can be evaluated by a typically sized PC (Pentium processor, 32 MB of memory). The current advances in PC technology and the refinement of deductive analysis algorithms in the DFM analysis engine will allow for more model complexity. Practical computational considerations that will permit a relatively large number of states in DFM models include:

- Increased computational efficiency within the 32 MB RAM solution environment.

- Expansion of the computational efficiency and capability at the DFM algorithmic level by prioritizing and minimization of RAM usage.
- Expansion of computer capability by writing the DFM code to utilize the Hard Drive (HD) memory.

It is believed that the use of such techniques will prevent any limitations on the complexity of team related accident scenarios that can be evaluated, within the DFM structure presented in this document.

A final point to be made here has to do with the quantification of probabilities. Obviously, the final goal of an analysis using the ATHEANA methodology and the DFM framework is the calculation of probabilities for human failure events. Although probability quantification is an issue that was not specifically addressed in the research described in this report, some observations on it can be made here. First, once the combinations of unsafe acts and error forcing contexts that lead to a human failure event have been identified, their probabilities of occurrence can be calculated using the techniques of ATHEANA, or some other technique. From this point of view, the important thing is that they were identified in the first place. Second, if a probabilistic version of DFM (yet to be developed, but on the drawing board) is used, the combinations of unsafe actions and error forcing contexts can be identified and quantified in the same analysis. Furthermore, the probability of the human failure event that is the 'top event' can be quantified at the same time. Note that this quantification would implicitly consider any dependencies and common causes inherent in the problem; all that is required is that these be incorporated into the DFM model in the first place. This is food for thought; it may be that probabilistic analysis is where the DFM approach may really shine.

9. REFERENCES

Barnes, V. E., R. J. Mumaw, and I. Schoenfeld: "Communication Errors in Nuclear Power Plants," *Proceedings of the 1996 ANS International Topical Meeting on Nuclear Power Plant Instrumentation, Control, and Human-Machine Interface Technologies*, 671-678, 1996.

Barriere, M. T., W. J. Luckas, D. W. Whitehead and A. M. Ramey-Smith,: "An Analysis of Operational Experience During LP&S and a Plan for Addressing Human Reliability Assessment Issues," NUREG/CR-6093, 1994.

Barriere, M.T. Forester, J., Bley, D., Cooper, S., Kolaczkowski, A., Luckas, W., Parry, G., Ramey-Smith, A., Thompson, C., Whitehead D., and Wreathall, J. "Technical Basis and Implementation Guidelines for A Technique for Human Event Analysis (ATHEANA)", Draft Report for Comment, NUREG-1624, May 1998.

Cacciabue, P.C.: "Dynamic vs Static Reliability Approaches: A Comparison of HERMES and THERP on a Nuclear Study Case," ANS Probabilistic Safety Assessment Topical Meeting, Park City Utah, September 1996.

Cooper, S.E. , J Wreathall, C.M. Thompson, M. Drouin, D.C. Bley, E. M. Roth, G.W Parry, W.J. Lucas: "Knowledge Base For The Human Reliability Analysis Method, A Technique For Human Error Analysis (ATHEANA)," ANS Probabilistic Safety Assessment Topical Meeting, Park City Utah, September 1996.

Dang, V. N and N.I. Siu: "The OPSIM Dynamic Operator Model; Implementation and Some Results," ANS Probabilistic Safety Assessment 96 Topical Meeting, Park City Utah, September 1996.

Garrett, C., S. Guarro and G. Apostolakis: "The Dynamic Flowgraph Methodology for Assessing the Dependability of Software Systems," *IEEE Transactions on Systems, Man and Cybernetics*, **25**, pp. 824-840, 1995a.

Garrett, C., M. Yau, S. Guarro and G. Apostolakis: "Assessing the Dependability of Embedded Software Systems Using the Dynamic Flowgraph Methodology," in *Dependable Computing and Fault-Tolerant Systems Vol. 9*, F. Cristian, G. Le Lann, T. Lunt (eds.), Springer-Verlag, Wien, 1995b.

Garriba, S., E. Guagnini and P. Mussio: "Multiple-Valued Logic Trees: Meaning and Prime Implicants," *IEEE Transactions on Reliability*, **R-34**, 463-472, 1985.

Gertman, D. I., L. N. Haney and N. O. Siu: "Representing Context, Cognition and Crew Performance in a Shutdown Risk Assessment," *Reliability Engineering and System Safety*, **52**, 261-278, 1996.

Guarro, S. B. and D. Okrent: "The Logic Flowgraph: A New Approach to Process Failure Modeling and Diagnosis for Disturbance Analysis Applications," Nuclear Technology 67, 148-359, 1984.

Guarro, S., M. Yau and M. Motamed: Development of Tools for Safety Analysis of Control Software in Advanced Reactors, U.S. Nuclear Regulatory Commission Report NUREG/CR-6465, 1996.

Henley, E.J. and H. Kumamoto: *Probabilistic Risk Assessment: Reliability Engineering, Design, and Analysis*, IEEE Press, 1992.

Julius, J., E. Jorgenson, G. W. Parry and A. M. Mosleh: "A Procedure for the Analysis of Errors of Commission in a Probabilistic Safety Assessment of a Nuclear Power Plant at Full Power," *Reliability Engineering and System Safety*, **50**, 189-201, 1995.

Mott, T.H.: "Determination of the Irredundant Normal Forms of a Truth Function by Iterated Consensus of the Prime Implicants," *IEEE Transactions on Electronic Computers*, **9**, pp. 245-252, 1960.

Ogunbiyi, I.E.: *Application of Decision Tables to Risk Analysis Studies*, Ph.D. Dissertation, University of Houston, TX, 1980.

Ogunbiyi, I. E. and E. J. Henley: "Irredundant Forms and Prime Implicants of a Function with Multistate Variables," *IEEE Transactions on Reliability*, **R-30**, 39-42, 1981.

Parry, G.W. , J. Wreathall, D.C. Bley, W.J Lucas, J.H. Taylor, S.E. Cooper, C.M Thompson, A.M. Ramey-Smith:. "A Process For Application Of ATHEANA- A New Method," ANS Probabilistic Safety Assessment 96 Topical Meeting, Park City Utah, September 1996.

Quine, W.V.: "The Problem of Simplifying Truth Functions," *American Mathematical Monthly*, **59**, pp. 521-531, 1952.

Quine, W.V.: "A Way to Simplify Truth Functions," *American Mathematical Monthly*, **62**, pp. 627-631, 1955.

Reason, J. T.,: *Human Error*, Cambridge University Press, Cambridge, MA, 1990.

Roth, E. M., R. J. Mumaw and P. M. Lewis: "An Empirical Investigation of Operator Performance in Cognitively Demanding Simulated Emergencies," NUREG/CR-6208, 1994.

Salem, S.L., G.E. Apostolakis and D. Okrent: "A New Methodology for the Computer-Aided Construction of Fault Trees," *Annals of Nuclear Energy*, **4**, pp. 417-433, 1977.

Salem, S.L., J.S. Wu and G.E. Apostolakis: "Decision Table Development and Application to the Construction of Fault Trees," *Nuclear Technology*, **42**, pp. 51-64, 1979.

Shields, E. J., G. E. Apostolakis and S. B. Guarro: "Determining the Prime Implicants for Multi-State Embedded Systems," *Proceedings of PSAM-II*, San Diego, CA, 1994.

Yau, M.: *Dynamic Flowgraph Methodology for the Analysis of Software Based Controlled Systems*, Ph.D. Dissertation, University of California, Los Angeles, 1997.

APPENDIX A. DESCRIPTION OF DEVELOPMENT TEST CASE MODEL

Appendix A presents the nodes and decision tables for the development case study, introduced in Section V.4.2.

A.1 Node RL

Node RL represents the coolant level in the RCS. In this example, three states for the RCS level is an absolute minimum. There must be a state for an RCS level of greater than 21 inches in the Hot Leg, in which case there is adequate core cooling and the operators are not instructed to recover the RCS inventory (unless the RCS level is decreasing), i. e., the core cooling safety function is not threatened. There must also be a state for an RCS level of less than 21 inches in the Hot Leg but greater than the level of the top of the fuel, in which case the operators are instructed to recover RCS inventory, but there is still adequate core cooling, i. e., the core cooling safety function is adequate but threatened. The remaining state is the RCS level for which the core is uncovered. This last state is the undesirable state that would be used for a top event. Thus the states of Node RL are shown in Table A.I.

Table A.I: States for node RL

State	Description
0	adequate - greater than 21" in the hot leg
1	barely adequate - less than 21" in the hot leg but above the top of the active fuel
2	inadequate - below the top of the active fuel

A.2 Node RC

Node RC represents the rate of change of the RCS level. Node RC has three states, shown in Table A.II.

Table A.II: States for node RC

State	Description
0	decreasing
1	stable
2	increasing

A.3 Transition Box TT1

Transition box TT1 models how the RCS level at the next time step depends on the current RCS level and the current rate of change of the RCS level. The decision table for transition box TT1 is shown in Table A.III.

Table A.III: Decision table for transition box TT1

Node RL		Node RC		Node RL	
State	Description	State	Description	State	Description
0	inadequate	0	decreasing	0	inadequate
0	inadequate	1	stable	0	inadequate
0	inadequate	2	increasing	0	inadequate
0	inadequate	2	increasing	1	barely adequate
1	barely adequate	0	decreasing	0	inadequate
1	barely adequate	0	decreasing	1	barely adequate
1	barely adequate	1	stable	1	barely adequate
1	barely adequate	2	increasing	1	barely adequate

A.4 Node RLS

Node RLS represents the state of the RCS level sensor. Although there are three systems for RCS level monitoring, the RWLP system, the HJTC system and the RWLI system, we assume for simplicity in this example that there is

only one sensor. It is straightforward to model three sensors, although that may result in very large decision tables. Since node RLS represents a hardware state, it is, by nature, a discrete node. However, in addition to the normal reliable state of the sensor, in which the value of the RCS level indicated is the actual value, we postulate two more failed states. One state is the case in which the sensor is completely broken (the 'failed' state) and does not provide any reading at all (or a reading of zero). The other state is the case in which the sensor provides an unreliable reading (the 'unreliable' state). The states of node RLS are shown in Table A.IV.

Table A.IV: States for Node RLS

State	Description
0	normal
1	failed
2	unreliable

A.5 Node RCI

Node RCI represents the indication of the rate of change of the RCS level. Node RCI has the same states as node RC with the addition of the no reading state, as shown in Table A.V.

Table A.V: States for Node RCI

State	Description
0	increasing
1	decreasing
2	stable
3	no reading

A.6 Node RLI

Node RLI represents the indication of the RCS level. The states for node RLI are the same as those for node RL with the addition of the no reading state, as shown in Table A.VI.

Table A.VI: States for Node RLI

State	Description
0	adequate
1	barely adequate
2	inadequate
3	no reading

A.7 Transfer Box T1

Transfer box T1 models the dependence of the RCS level sensor's indication of the RCS level change on the actual RCS level change and the status of the sensor. The decision table for transfer box T1 is shown Table A.VII. If the status of the sensor is normal, the indication is the actual value. If the sensor is failed, there is no indication. The possibility is which the instrument provides the wrong indication is not considered.

Table A.VII: Decision table for transfer box T1

RL		RLS		RLI	
State	Description	State	Description	State	Description
2	adequate	0	normal	2	adequate
1	barely adequate	0	normal	1	barely adequate
0	inadequate	0	normal	0	inadequate
-	-	1	failed	3	no reading

A.8 Transfer Box T2

Transfer box T2 models the dependence of the RCS level sensor's reading of the RCS level on the actual RCS level and the status of the sensor. The decision table for transfer box T1 is shown Table A.VIII. If the status of the sensor is normal, the reading is the actual value. If the sensor is failed, there is no reading.

Table A.VIII: Decision table for transfer box T2

RC		RLS		RCI	
State	Description	State	Description	State	Description
0	decreasing	0	normal	0	decreasing
1	stable	0	normal	1	stable
2	increasing	0	normal	2	increasing
-	-	1	failed	3	no reading

A.9 Node IG

Node IG represents the information gathering process of the operator monitoring the RCS level. This node is used to model errors in information gathering. The possible errors considered here include: the operator misreads the RCS level, the operator misinterprets the RCS level change and the operator does not read the sensor. The States for node IG are shown in Table A.IX.

Table A.IX: States for Node IG

State	Description
0	operator reads instrument correctly
1	operator does not read instrument

A.10 Node RCA

Node RC represents the operator's current assessment of the rate of change of the RCS level. In addition to the three states of the corresponding plant parameter node (node RC), this note has a state that represents that the operator is unsure whether the level is increasing, decreasing or stable. Even though there are three possible uncertain states, for now we will only consider the state that represents complete uncertainty. The states for node RCA are shown in Table A.X below.

Table A.X: States for Node RCA

State	Description
0	RCS level is increasing
1	RCS level is stable
2	RCS level is decreasing
3	Uncertain

A.11 Node RLA

Node RLA represents the operator's current assessment of the coolant level in the RCS. In addition to the three states of node RL, node RLA also has a state representing complete uncertainty. Thus the states of Node RLA are shown in Table A.XI.

Table A.XI: States for Node RL

State	Description
0	adequate - greater than 21" in the hot leg
1	barely adequate - less than 21" in the hot leg but above the top of the active fuel
2	inadequate - below the top of the active fuel
3	completely uncertain

A.12 Transfer Box T3

Transfer Box T3 models the dependence of the SRO's assessment of the RCS level change on the actual indication, his information gathering process and his prediction of what the RCS level change should be. The decision table for transfer box T3 is shown in Table A.XII.

If the operator reads the instrument correctly, then his assessment is exactly the same the actual indication. If the operator neglects to read the indication, or if there is no indication, then his assessment will be the same as his predicted value.

Table A.XII: Decision table for transfer box T3

RCI		RCP		IG		RCA	
State	Description	State	Description	State	Description	State	Description
0	decreasing	-	-	0	no error	0	decreasing
-	-	0	decreasing	1	does not read	0	decreasing
-	-	1	stable	1	does not read	1	stable
-	-	2	increasing	1	does not read	2	increasing
-	-	3	uncertain	1	does not read	3	unknown
1	stable	-	-	0	no error	1	stable
2	increasing	-	-	0	no error	2	increasing
3	no reading	0	decreasing	-	-	0	decreasing
3	no reading	1	stable	-	-	1	stable
3	no reading	2	increasing	-	-	2	increasing
3	no reading	3	unknown	-	-	3	unknown

A.13 Transfer box T4

Transfer Box T4 models the dependence of the operators assessment of the RCS level on the indicated RCS level, his information gathering process and his prediction of what the RCS level should be. The decision table for transfer box T4 is shown in Table A.XIII.

If the operator reads the instrument correctly, then his assessment is exactly the same the actual indication. If the operator neglects to read the indication, or if there is no indication, then his assessment will be the same as his predicted value.

Table A.XIII: Decision table for transfer box T4

RLI		RLP		IG		RLA	
State	Description	State	Description	State	Description	State	Description
0	inadequate	999		0	correct	0	inadequate
-		0	inadequate	1	does not read	0	inadequate
-	-	1	barely adequate	1	does not read	1	barely adequate
-	-	2	adequate	1	does not read	2	adequate
-	-	3	unknown	1	does not read	3	unknown
1	barely adequate	-	-	0	correct	1	barely adequate
2	adequate	-	-	0	correct	2	adequate
3	no reading	0	inadequate	-	-	0	inadequate
3	no reading	1	inadequate	-	-	1	inadequate
3	no reading	2	adequate	-	-	2	adequate
3	no reading	3	unknown	-	-	3	unknown

A.14 Node RCP

Node RCP represents the operator's prediction of what the RCS level change will be in the next time step. This node depends on other nodes that are not shown in Figure 1. It has the same states of node RCA, as shown in Table A.XIV.

Table A.XIV: States for Node RCP

State	Description
0	increasing
1	decreasing
2	stable
3	uncertain

A.15 Node RLP

Node RLP represents the operator's assessment of what the RCS level will be in the next time step. The states of node RLP are shown in Table A.XV. In general, the operator's prediction of what the RCS level will be in the future depends on his current assessment of the RCS level and his current assessment of the rate of change of the RCS level.

Table A.XV: States for Node RLP

State	Description
0	adequate
1	barely adequate
2	inadequate
3	uncertain

A.16 Transition Box TT2

Transition box TT2 represents how the operator's prediction of what the RCS level will be in the next time step depends on his current assessment of the RCS level (node RLA) and whether it is increasing or decreasing (node RCA). The decision table for transition box TT2 is shown in Table A.XVI.

Since it takes more than one time step for the RCS level to traverse a state if the level is changing, this decision table is also non-deterministic. If the RCS level is changing, the SRO's prediction of the RCS level for the next time step may be a state higher or lower, or it may stay in the same state. This is because his current assessment may or may not be at the edge of the range of RCS level values defined by that state. If the SRO's has an uncertain assessment

of the RCS level change, then his predicted RCS level could either stay at his current RCS level assessment or become uncertain. Again, this is because his current assessment may or may not be at the edge of the range of RCS level values defined by that state. If the SRO's has an uncertain current RCS level assessment, then the predicted RCS level is always uncertain.

Table A.XVI: Decision table for transition box TT2

Node RLA		Node RCA		Node RLP	
State	Description	State	Description	State	Description
0	inadequate	0	decreasing	0	inadequate
0	inadequate	1	stable	0	inadequate
0	inadequate	2	increasing	0	inadequate
0	inadequate	2	increasing	1	barely adequate
0	inadequate	3	uncertain	0	adequate
0	inadequate	3	uncertain	3	uncertain
1	barely adequate	0	decreasing	0	inadequate
1	barely adequate	0	decreasing	1	barely adequate
1	barely adequate	1	stable	1	barely adequate
1	barely adequate	2	increasing	1	barely adequate
1	barely adequate	2	increasing	2	adequate
1	barely adequate	3	uncertain	1	barely adequate
1	barely adequate	3	uncertain	3	uncertain
2	adequate	0	decreasing	1	barely adequate
2	adequate	0	decreasing	2	adequate
2	adequate	1	stable	2	adequate
2	adequate	2	increasing	2	adequate
2	adequate	3	uncertain	1	barely adequate
2	adequate	3	uncertain	3	uncertain
3	uncertain	-	-	3	uncertain

A.17 Node AS1

Node AS1 represents the SRO's perception of the current activity status, i.e., whether or not RCS inventory recovery actions have been initiated. The states for node AS1 are given in Table A.XVII below. If node AS1 is in state 2, the SRO is unsure whether or not the procedures have been initiated, either because the RO failed to confirm the fact, or because the SRO has received conflicting information from the sensors; i.e., he believes that the RCS level is not increasing.

Table A.XVII: States for Node AS1

State	Description
0	Recover RCS inventory procedures initiated
1	Recover RCS inventory procedures not initiated
2	unknown

A.18 Transition Box TT3

Transition Box TT3 models the dependence of the SRO's prediction of what the RCS level change will be in the next time step on his current assessment and his current perception of the status of RCS inventory recovery activities. The decision table for Transition Box TT3 is given in Table A.XVIII below.

For the case in which the SRO believes RCS inventory recovery procedures have been initiated, he will always believe the RCS level should be increasing in the next time step, regardless of his current assessment of the current RCS level change. For the case in which the SRO believes RCS inventory recovery procedures have not been initiated, he will believe the RCS level in the next time step should be the same as his current assessment. For the case in which the SRO doesn't know if RCS inventory recovery procedures have been initiated, he be uncertain

what the RCS level should be in the next time step, unless his current assessment is that the RCS level is increasing. In that case he will believe that the RCS level should be increasing in the next time step.

Table A.XVIII: Decision table for transition box TT3

Node RCA		Node AS1		Node RCP	
State	Description	State	Description	State	Description
-	-	0	initiated	2	increasing
0	decreasing	1	not initiated	0	decreasing
0	-	2	unknown	3	uncertain
1	stable	1	not initiated	1	stable
1	stable	2	unknown	3	uncertain
2	increasing	-	-	2	increasing
3	uncertain	1	not initiated	3	uncertain
3	uncertain	2	unknown	3	uncertain

A.19 Node D1

Node D1 represents the SRO's decision of whether or not to initiate RCS inventory recovery procedures. Node D1 can assume two states reflecting these two possible decisions: "initiate RCS recovery" and "do not initiate RCS recovery", as shown in Table A.XIX.

Table A.XIX: States for Node D1

State	Description
0	initiate recover RCS inventory procedures
1	do not initiate recover RCS inventory procedures

The decision actually made depends on the SRO's current assessment of the plant state, in this case his assessment of the RCS level (node RLA), and whether or not it is decreasing (node RCA). It also depends on node RP1, which represents root causes of errors in response planning.

A.20 Node RP1

Node DP1 represents root causes of errors in response planning. In this case only consider the following root cause: operator disregards procedures. In this case, if the level is relatively high, but is decreasing or if the level barely adequate and stable, the operator might decide that it is not necessary to recover RCS inventory at that time. The states of node DP1 are given in Table A.XX below.

There is a third possibility in the case that the operator is uncertain of the RCS level or whether or not in is decreasing. In such a case conventional wisdom would tell us that the operator should initiate recovery of RCS inventory procedures anyway. However, the operators uncertainty may cause him to postpone the decision until a later time, hoping he resolves his uncertainty

Table A.XX: States for Node DP1

State	Description
0	no error
1	operator disregards procedures

A.21 Transfer Box T5

The decision table for transfer box T5 is given in Table A.XXI. If the SRO's believes the RCS level is increasing, then he must believe RCS inventory recovery procedures have been initiated, and will obviously not decide to initiate them. If he believes that the RCS level is inadequate, then he will always decide to initiate RCS inventory recovery procedures (unless he believes the RCS level is increasing).

If the SRO believes the RCS level is barely adequate and decreasing, he will always decide to initiate RCS inventory recovery procedures. If the SRO believes the RCS level is barely adequate and stable, what he decides to do

depends on the state of node RP1. If he does not make a response planning error, he will decide to initiate the procedures. However, if node RP1 is in the 'disregards' state, then the SRO will decide not to initiate the procedures right away. If the SRO believes the RCS level is barely adequate and is uncertain if it is decreasing or not, he will always decide to initiate RCS inventory recovery procedures.

If the SRO believes the RCS level is adequate and decreasing, what he decides to do depends on the state of node RP1. If he does not make a response planning error, he will decide to initiate the procedures. However, if node RP1 is in the 'disregards' state, then the SRO will decide not to initiate the procedures right away. If the SRO believes the RCS level is adequate and stable, he will always decide not to initiate recovery procedures. If the SRO believes the RCS level is adequate and is uncertain whether or not the RCS level is decreasing, what he decides to do depends on the state of node RP1. If he does not make a response planning error, he will decide to initiate the procedures. However, if node RP1 is in the 'disregards' state, then the SRO will decide not to initiate the procedures right away. If the SRO is uncertain about the RCS level, he will always decide to initiate recovery procedures.

Table A.XXI: Decision table for transfer box T5

RLA		RCA		RP1		D1	
State	Description	State	Description	State	Description	State	Description
0	inadequate	0	decreasing	-	-	0	initiate
0	inadequate	1	stable	-	-	0	initiate
-	-	2	increasing	-	-	1	do not initiate
0	inadequate	3	uncertain	-	-	0	initiate
1	barely adequate	0	decreasing	-	-	0	initiate
1	barely adequate	1	stable	0	no error	0	initiate
1	barely adequate	1	stable	1	disregards	1	do not initiate
1	barely adequate	3	uncertain	-	-	0	initiate
2	adequate	0	decreasing	0	no error	0	initiate
2	adequate	0	decreasing	1	disregards	1	do not initiate
2	adequate	1	stable	-	-	1	do not initiate
2	adequate	3	uncertain	0	no error	0	initiate
2	adequate	3	uncertain	1	disregards	1	do not initiate
3	uncertain	-	-	-	-	0	initiate

A.22 Node CD1

Node CD1 represents the decision made by the SRO regarding whether or not to initiate recovery of RCS inventory actions, as communicated to the RO responsible for actually implementing the actions. Thus, the states of node CD1 are the same as those for node D1, given in Table V.XIX.

A.23 Node CP1

Node CP1 represents root causes of communication errors between the two control room operators. Node CP1 has the states shown in Table A.XXII.

Table A.XXII: States for Node CP1

State	Description
0	no error
1	error

A.24 Transfer Box T6

The decision table for transfer box T6 is given in Table A.XXIII below. If the SRO has decided to initiate recovery of RCS inventory procedures and there is a communication error, the RO will not get the message (or will misunderstand it which amounts to the same thing), and will not initiate the procedures.

Table A.XXIII: Decision table for transfer box T6

Node D1		Node CP1		Node CD1	
State	Description	State	Description	State	Description
0	initiate	0	no error	0	initiate
0	initiate	1	error	1	do not initiate
1	do not initiate	-	-	1	do not initiate

A.25 Node D2

Node D2 represents the decision made by the RO of whether to use the HPSI system or the Containment Spray system for emergency coolant injection to recover RCS inventory. The states representing these decisions, along with a state for doing nothing are shown in Table A.XXIV.

The correct decision depends on the status of the two systems. If the HPSI system is unavailable, the operators are instructed to use the Containment spray system. Also, if the operators had previously decided not to initiate recovery of RCS inventory, they would not need to make the decision represented by node D2. Thus, to reflect this, node D2 depends on node CD1, and there is a "no injection" state for node D2.

Table A.XXIV: States for Node D2

State	Description
0	use HPSI system
1	use Containment Spray system
2	no injection

A.26 Node DP2

Node DP2 represents root causes in response planning for the decision of which system to use for RCS inventory recovery. One possible error is considered here. If the HPSI system is unavailable, the operator may believe that the system will be recovered soon and, therefore, defer the use of the Containment spray system. Node DP2 has the states given in Table A.XXV below.

Table A.XXV: States for Node DP2

State	Description
0	no error
1	operator forestalls use of containment spray

A.27 Transfer Box T7

The decision table for transfer box T7 is given in Table A.XXVI below.
If the RO believes the HPSI system is already operating, he will decide to do nothing (the 'neither' state of D2). Similarly, if the RO believes the Containment Spray system is already operating, he will decide to do nothing.
If the operator believes the HPSI system is available, he will attempt to use it to recover RCS inventory unless he believes the containment spray system is already operating. If the operator believes the HPSI system is unavailable, but the Containment Spray system is available, he will decide to use the Containment Spray system, unless he forestalls because he thinks the HPSI system may become available again soon. If the operator believes both systems are unavailable, he will not attempt to recover RCS inventory using either system.

Table A.XXVI: Decision Table for transfer box T7

CD1		HPS		CSS		DP2		D2	
State	Description	State	Description	State	Description	State	Description	State	Description
0	initiate	0	operating	-	-	-	-	2	neither
0	initiate	-	-	0	operating	-	-	2	neither
0	initiate	1	available	1	available	-	-	0	use HPSI
0	initiate	1	available	2	unavailable	-	-	0	use HPSI
0	initiate	2	unavailable	1	available	0	no error	1	use CSS
0	initiate	2	unavailable	1	available	1	forestalls	2	neither
0	initiate	2	unavailable	2	unavailable	-	-	2	neither
1	do not initiate	-	-	-	-	-	-	2	neither

A.28 Node CD2

Node CD2 represents the confirmation communicated back to the SRO form the RO about whether or not he took the actions he was asked to. The states of node CD2 are shown in Table A.XXVII. State 1 ('not confirmed') indicates that the SRO did not receiver confirmation from the RO that the recovery of RCS inventory was initiated. This could be either because the SRO had never communicated to the RO to initiate recovery of RCS inventory procedures in the first place, because the RO never issued a confirmation even though he did initiate the procedures, or because the SRO did not hear the confirmation.

Table A.XXVII: States for Node CD2

State	Description
0	confirmed
1	not confirmed

A.29 Node CP2

Node CP2 represent errors that could occur during the confirmation communication process. The states of node CP2 are given in Table A.XXVIII.

Table A.XXVIII: States for Node CP2

State	Description
0	no error
1	error

A.30 Transition Box TT4

The decision table for transition box TT4 is shown in Table A.XXIX. The action is confirmed (state 0 of node CD2) if the RO decides to use the HPSI system of the containment spray system (states 0 and 1 of node D2) and no communication error is made (state 0 of node CP2).

Table A.XXIX: Decision table for transfer box TT4

Node D2		Node CP2		Node CD2	
State	Description	State	Description	State	Description
0	use HPSI	0	no error	0	confirmed
0	use HPSI	1	error	1	not confirmed
1	use CSS	0	no error	0	confirmed
1	use CSS	1	error	1	not confirmed
2	neither	-	-	1	not confirmed

A.31 Transfer Box T8

The states for transfer box T8 are shown in Table A.XXX. If the initiation of RCS inventory recovery procedures have been confirmed by the RO, but the SRO believes the RCS level is decreasing or stable, he will become uncertain about whether or not the procedures have been initiated (the RO may have made a mistake, etc.). If the SRO believes the RCS level is increasing, then he will believe the procedures have been initiated regardless if the RO has confirmed or not (it is the only explanation for the increase in the RCS level). If the initiation of RCS inventory recovery procedures have been confirmed by the RO, but the SRO is uncertain about the RCS level, he will believe that the procedures have been initiated (he has no reason not to).

If the initiation of RCS inventory recovery procedures have not been confirmed by the RO (either because he was not instructed to initiate the procedures or because of a communication error), and the SRO believes the RCS level is either decreasing or stable, then he has no reason to believe anything but that RCS inventory recovery procedures have not been initiated. If the initiation of RCS inventory recovery procedures have not been confirmed by the RO and the SRO is uncertain about the RCS level, he will continue to be uncertain whether or not RCS inventory procedures have been initiated.

Table A.XXX: Decision table for transfer box TT8

Node RCA		Node CD2		Node AS1	
State	Description	State	Description	State	Description
0	decreasing	0	confirmed	2	unknown
0	decreasing	1	not confirmed	1	not initiated
1	stable	0	confirmed	2	unknown
1	stable	1	not confirmed	1	not initiated
2	increasing	-	-	0	initiated
3	uncertain	0	confirmed	0	initiated
3	uncertain	1	not confirmed	2	unknown

A.32 Node HPS

Node HPS represents the status of the HPSI system and has three states: 'operating', 'available' and 'unavailable', as shown in Table V.XXXI.

Table A.XXXI: States for Node HPS

State	Description
0	operating
1	available
2	unavailable

A.33 Node RI1

Node RI1 represents the root causes of failures in response implementation for the action of starting the HPSI system. There are two states: 'slip' and 'no slip', as shown in Table A.XXXII. The 'slip' state models the case in which the operator makes a mistake in the performance of the action, in this case, starting the HPSI system.

Table A.XXXII: States for Node RI1

State	Description
0	slip
1	no slip

A.34 Transition Box TT5

The decision table for transition box TT5 is shown in Table A.XXXIII. If the decision is to use the HPSI system for emergency coolant injection, then the HPSI system will be operating state, unless it is unavailable, or the operator makes an error in response implementation. Otherwise, the status of the HPSI system remains the same.

Table A.XXXIII: Decision Table for Transition Box TT5

D2		RI1		HPSP		HPS	
State	Description	State	Description	State	Description	State	Description
0	use HPSI	0	no slip	0	operating	0	operating
0	use HPSI	0	no slip	1	available	0	operating
0	use HPSI	0	no slip	2	unavailable	2	unavailable
0	use HPSI	1	slip	0	operating	0	operating
0	use HPSI	1	slip	1	available	1	available
0	use HPSI	1	slip	2	unavailable	2	unavailable
1	use CSSI	-	-	0	operating	0	operating
1	use CSSI	-	-	1	available	1	available
1	use CSSI	-	-	2	unavailable	2	unavailable
2	use neither	-	-	0	operating	0	operating
2	use neither	-	-	1	available	1	available
2	use neither	-	-	2	unavailable	2	unavailable

A.35 Node HPF

Node HPF represents the emergency coolant flow from the HPSI system. It has two states: 'flow' and 'no flow' as shown in Table A.XXXIV.

Table A.XXXIV: States for Node HPF

State	Description
0	flow
1	no flow

A.36 Transfer Box T9

Table A.XXXV shows the decision table for transfer box T9. There is emergency coolant flow from the HPSI system only if the HPSI system is operating.

Table A.XXXV: Decision table for transfer box T9

HPS		HPF	
State	Description	State	Description
0	operating	0	flow
1	available	1	no flow
2	unavailable	1	no flow

A.37 Node CD3

As mentioned in previous reports, if the RO decides to use the containment spray system for emergency coolant injection, then he must notify a remote operator of his decision and instruct the remote operator to open the appropriate valve. Node CD3 represents whether or not the communication is successful, and thus has two states: 'align valve' and 'do not align valve', as shown in Table A.XXXVI. The 'do not start CSS' state includes the situation in which the RO decides to use the HPSI system for emergency coolant injection or decides to do nothing, and thus does not attempt a communication with the remote operator.

Table A.XXXVI: States for node CD3

State	Description
0	align valve
1	do not align valve

A.38 Node CP3

Node CP3 represents root causes of communication errors between the RO and the remote operator. The states of node CP3 are shown in Table A.XXXVII.

Table A.XXXVII: States for Node CP3

State	Description
0	no error
1	error

A.39 Transfer Box T10

As mentioned above, the unsuccessful state of node CD3 includes the case in which no communication is attempted. Thus, the decision table for transfer box T10 takes the form shown in Table A.XXXVIII. The only case in which node CD3 gets the 'align valve' state is when the RO has decided to use the Containment Spray system (state C of node D2) and there are no communication errors (state N of node CP3).

Table A.XXXVIII: Decision table for transfer box T10

D2		CP3		CD3	
State	Description	State	Description	State	Description
0	use HPSI	-	-	1	do not align valve
1	use CSS	0	no error	0	align valve
1	use CSS	1	error	1	do not align valve
2	neither	-	-	1	do not align valve

A.40 Node CSS

Node CSS represents the status of the containment spray system and has three states: 'operating', 'available' and 'unavailable', as shown in Table A.XXXIX.

Table A.XXXIX: States for Node CSS

State	Description
0	operating
1	available
2	unavailable

A.41 Node RI2

Node RI2 represents the root causes of failures in response implementation for the action of starting the Containment Spray system. There are two states: 'slip', and 'no slip', as shown in Table A.XL. The slip' state models both the case in which the RO makes a mistake in starting the Containment Spray system and the case in which the remote operator makes a mistake in aligning the valve.

Table A.XL: States for Node RI2

State	Description
0	no slip
1	slip

A.42 Transition Box TT6

The decision table for transition box TT6 is shown in Table A.XLI. If the decision is to use the Containment Spray system for emergency coolant injection, and the remote operator is properly instructed to align the valve, then the Containment Spray system will be operating state, unless it is unavailable, or the operator makes an error in response implementation. Otherwise, the status of the Containment Spray system remains the same.

Table A.XLI: Decision Table for Transition Box TT6

CD3		CSSP		RI2		CSS	
State	Description	State	Description	State	Description	State	Description
0	align valve	0	operating	-	-	0	operating
0	align valve	1	available	0	no slip	0	operating
0	align valve	1	available	1	slip	1	available
-	-	2	unavailable	-	-	2	unavailable
1	do not align valve	0	operating	-	-	0	operating
1	do not align valve	1	available	-	-	1	available
1	do not align valve	2	unavailable	-	-	2	unavailable

A.43 Node CSF

Node CSF represents the emergency coolant flow from the Containment Spray system. There are two states: 'flow' and 'no flow' as shown in Table A.XLII.

Table A.XLII: States for node CSF

State	Description
0	flow
1	no flow

A.44 Transfer Box T11

The decision table for transfer box T1 is shown in Table A.XLIII below. If the Containment Spray system is operating (state O of node CSS) then there is flow (state F of node CSF). Otherwise there is no flow (state N of node CSF).

Table A.XLIII: Decision table for transfer box T11

CSS		CSF	
State	Description	State	Description
0	operating	0	flow
1	available	1	no flow
2	unavailable	1	no flow

A.45 Node L

Node L models whether or not there is a leak in the RCS. The states for node L are shown in Table A.XLIV.

Table A.XLIV: States for Node L

State	Description
0	no leak
1	leak

A.46 Transfer Box T12

Transfer box T12 models the dependence of the RCS level change on the flow from either the HPSI system or the Containment Spray system. The decision table for transfer box T12 is shown in Table A.XLV below.

Table A.XLV: Decision table for transfer box T12

HPF		CSF		L		RC	
State	Decision	State	Decision	State	Description	State	Decision
0	flow	-	-	-	-	2	increasing
1	no flow	0	flow	-	-	2	increasing
1	no flow	1	no flow	0	no leak	1	stable
1	no flow	1	no flow	1	leak	0	decreasing

NRC FORM 335 (2-89) NRCM 1102, 3201, 3202

U.S. NUCLEAR REGULATORY COMMISSION

BIBLIOGRAPHIC DATA SHEET

(See instructions on the reverse)

1. REPORT NUMBER (Assigned by NRC, Add Vol., Supp., Rev., and Addendum Numbers, if any.)

NUREG/CR-6710

2. TITLE AND SUBTITLE

Extending the Dynamic Flowgraph Methodology (DFM) to Model Human Performance and Team Effects

3. DATE REPORT PUBLISHED

MONTH	YEAR
MARCH	2001

4. FIN OR GRANT NUMBER

W6719

5. AUTHOR(S)

Anthony Milici, Robert Mulvihill and Sergio Guarro

6. TYPE OF REPORT

7. PERIOD COVERED *(Inclusive Dates)*

10/01/96 to 12/31/98

8. PERFORMING ORGANIZATION - NAME AND ADDRESS *(If NRC, provide Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address; if contractor, provide name and mailing address.)*

ASCA, Inc.
704 Silver Spur Road
Rolling Hills Estates, CA 90274

9. SPONSORING ORGANIZATION - NAME AND ADDRESS *(If NRC, type "Same as above"; if contractor, provide NRC Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address.)*

Division of Systems Analysis and Regulatory Effectiveness
Office of Nuclear Regulatory Research
United States Nuclear Regulatory Commission
Washington, DC 20555-0001

10. SUPPLEMENTARY NOTES

J.J. Persensky, NRC Project Manager

11. ABSTRACT *(200 words or less)*

This report addresses the development of a structure for the modeling and analysis of control room teams to represent team related human errors of commission and omission in nuclear power plant accident scenarios. The structure includes the identification of unsafe actions (UAs) and error forcing contexts (EFCs) during abnormal or accident situations that can lead to a human failure event. This report also describes guidelines for the screening of sequences for which a dynamic flowgraph methodology (DFM) analysis could be effectively applied. The screening, DFM modeling and DFM analysis processes are demonstrated, via case studies, within the general human reliability analysis approach provided by the ATHEANA framework. In addition, this report describes extensions to the DFM methodology devised specifically to facilitate its application to teams effects modeling and analysis; in particular, it describes a library of pre-built DFM model modules that represent typical team-related cognitive, assessment and interaction processes.

12. KEY WORDS/DESCRIPTORS *(List words or phrases that will assist researchers in locating the report.)*

Dynamic Flowgraph Methodology, Control Room Teams, Human Error Analysis

13. AVAILABILITY STATEMENT

unlimited

14. SECURITY CLASSIFICATION

(This Page)

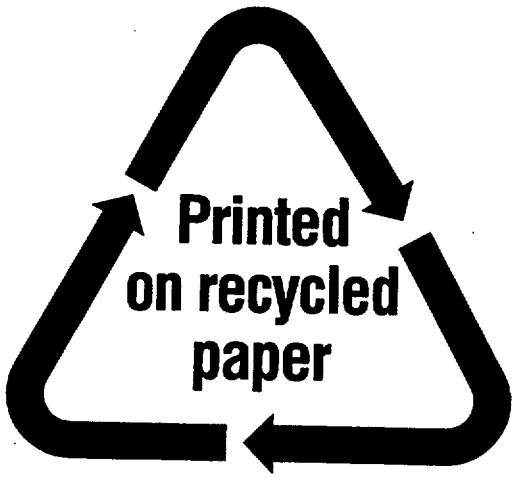
unclassified

(This Report)

unclassified

15. NUMBER OF PAGES

16. PRICE



Federal Recycling Program