CONF-810905--31

DE82 007346

INTERIM RELIABILITY EVALUATION PROGRAM BROWNS FERRY FAULT TREES

Milan E. Stewart

EG&G Idaho, Inc.
Idaho Falls, Idaho 83415, U.S.A.

ABSTRACT

An abbreviated fault tree method is used to evaluate and model Browns Ferry systems in the Interim Reliability Evaluation Programs, simplifying the recording and displaying of events, yet maintaining the system of identifying faults. The level of investigation is not changed. The analytical thought process inherent in the conventional method is not compromised. But the abbreviated method takes less time, and the fault modes are much more visible.

I. INTRODUCTION

As in the Reactor Safety Study,[1] fault tree analysis was used in the Interim Reliability Evaluation program (IREP) to evaluate nuclear plant systems. Fault trees developed were used as fault models to determine probabilities of systems failure and occurrences of accident sequences. However, whereas conventional fault trees[2] were constructed in the Reactor Safety Study, an abbreviated fault tree procedure was used in IREP to evaluate the Browns Ferry, Unit 1, plant. Neither the level of detail nor the logical thought process characteristic of the conventional method is compromised. The approach merely simplifies the manner of recording and displaying identified events. The abbreviated procedure has several distinct advantages over the conventional one, reducing the time and effort required to evaluate a system. The more important advantages can be summarized as follows:

- Fault trees are readily restructured for each new accident situation. Events can be quickly added or dropped, and blocks of events can be moved if the logic changes.
- Component fault modes and their logical relationship to system failure are more visible. A typical, conventionally developed system fault tree requires 20 to 30 large sheets of paper to show all component fault statements. These same statements can usually be shown on two or three 8-1/2 x 11-inch sheets. Reduced size and improved fault mode visibility make the trees much easier to check.
- A system evaluation is easier to stage. (Analysis staging is discussed in the final section.)
- The abbreviated procedure is more amenable to the treatment of common cause failures.

EAB

- In formal reports, most diagrams are replaced by tables, requiring less publication effort.

The abbreviated method is not new, but a natural evolvement of the conventional fault tree method used in the Reactor Safety Study. And it has been used in numerous systems reliability studies at the Idaho National Engineering Laboratory[3] and risk studies of the Clinch River Breeder Reactor[4] Plant and the Big Rock Point Plant.[5]

This paper is a summary of a fault tree guide prepared for use by the IREP Browns Ferry team. It discusses fault tree construction, with emphasis on the abbreviated method, component fault states, logic gates, event names, required conditions, and analysis staging—discussion that is apropos whether the conventional or abbreviated fault tree method is used.

2. SYSTEM FAILURE DEFINITION AND UNDESIRED EVENT

Fault tree analysis begins with a statement of an undesired event. Embodied in the statement must be the conditions that constitute failure of the system. For example, the undesired event, "insufficient coolant flow through the reactor core when the reactor is generating heat" is a complete logic statement specifying the requirements for reactor coolant. If a fault tree is developed about the undesired event, the analyst examines all systems—normal operating and emergency—that deliver coolant to the reactor vessel. He may define a more restricted undesired event, for example, "insufficient emergency coolant flow when normal flow is lost," where a fault tree is developed for the auxiliary coolant systems only. In any case, the top event, including conditions, must be compatible with the event tree sequence to which it pertains. Statements can be rather general. For example, the word insufficient implies that below some flow value the system will have failed. Where redundancy has been provided, however, the general statement must be made more specific to account for the redundant capabilities of the system. For example, the statement, "insufficient coolant flow . . . ," might be more specifically stated, "less than two-pump coolant flow . . . " (where more than two pumps exist). The fault tree is developed about the selected undesired event, and only events that relate logically to the occurrence of that undesired event are identified. Component failures that produce other undesired events when loss of flow is of concern (e.g., inadvertent operation of the system) will not be identified unless the particular component failures relate to the occurrence of both undesired events.

The undesired event and all subsequent events shown on the fault tree are binary. That is, if the event occurs the system (or component in more detailed parts of the tree) has failed; if the event does not occur the system has not failed. Ambiguous or "maybe" statements are not allowed on the tree. The statement is true if the event exists or false if the event does not exist.

3. FAULT TREE CONSTRUCTION

Once an undesired event is defined, a fault tree can be constructed about that event. A PWR high-pressure injection system (HPIS) will be used to illustrate the method. The top tiers of the fault tree will be constructed using the conventional method, then restructured using an abbreviated approach.

Figure 1 is a simplified schematic of the HPIS. It is used to provide emergency coolant to the reactor vessel in the event of a small loss-of-coolant accident when the reactor coolant system (RCS) is not depressurized sufficiently for core flood or low-pressure coolant injection. The HPIS is initiated automatically by an engineered safeguards actuation system (ESAS) upon 1500 psig decreasing RCS pressure or 4 psig increasing containment pressure. An ESAS signal will start the three pumps, open

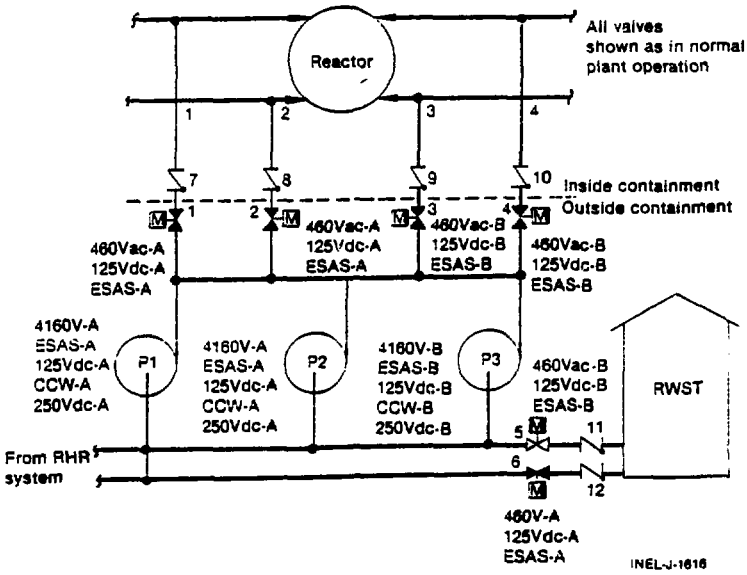

Figure 1. Simplified schematic of PWR high-pressure injection system.

refueling water storage tank (RWST) valve 6 (valve 5 is normally open), and open injection valves 1, 2, 3, and 4. All valves in connecting piping (not shown) are assumed to be closed for this example.

3.1 Conventional Fault Tree Construction

The undesired event selected for the HPIS must be compatible with the event tree sequence for which it applies. Suppose that a relief valve sticks open, heat removal through the power conversion system is lost, and the HPIS must provide emergency coolant to the reactor vessel. Suppose too, that one-pump HPIS flow through any path shown will suffice. An undesired, or top, event for the fault tree might be "less than one-pump HPIS flow to the RCS given a stuck-open relief valve and no heat removal through the power conversion system." Other top events would have been selected for other accident initiators and sequences, but this top event will illustrate the method. Since the "given" part of the undesired event statement specifies the conditions under which the fault events to be defined by the fault tree produce system failure (see Section 6), the top undesired event, as shown in the top rectangle, Figure 2, is translated into the two logic statements, (a) "stuck-open relief valve, no heat removal through power conversion system," (shown within a house symbol), and (b) "less than one-pump flow to the RCS," (shown within a rectangle). The house indicates the conditions under which "less than one-pump HPIS flow to the RCS" is a fault. The rectangle indicates a fault event that is developed further. Although not shown here, other conditions of the plant or system pertinent to the evaluation of the HPIS (e.g., no offsite power) should also be specified in the top event and house statements. Other house events are shown on subsequent tiers of the fault tree, indicating the normal operational state of components from which they transfer to a faulted state, unless these conditions are obvious.

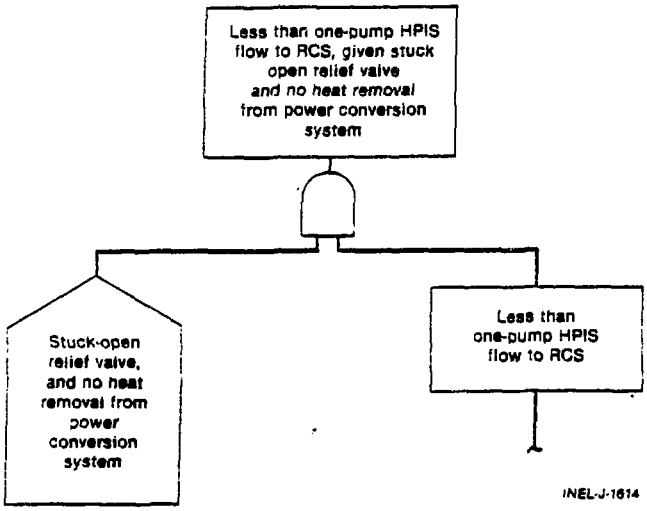

Figure 2. Top two fault tree tiers.

Next in the analysis is to translate the system event, "less than one-pump HPIS flow to the RCS," into subsystem fault statements. This can be done several ways, all of which should be logically equivalent. Figure 1 shows four redundant injection paths[6] (since the initiating event is a stuck-open relief valve, all paths are available), three redundant pump paths, two redundant pump suction paths, and a single RWST, so the event can be translated into the subsystem events in Figure 3. All the subsystem events relate to the system event by OR logic, since any one or more of the stated subsystem events will produce the system event. The subsystem events are further translated into individual path events. Figure 4 shows one subsystem event and the path events that cause it. The individual path fault events are input to an AND gate, since adequate flow can be achieved through any one path. The event in Figure 3, "insufficient water in the RWST," will not be expanded into its respective causes; so, the event is shown within a diamond.

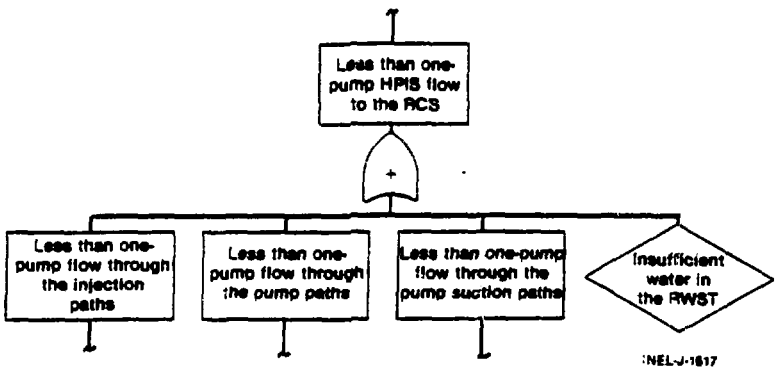

Figure 3. Translation of system event into subsystem events.

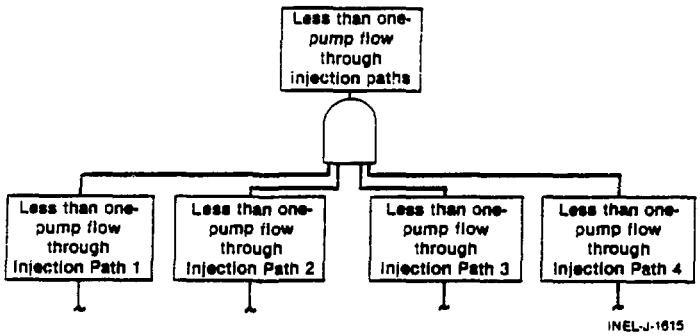

Figure 4. Translation of system event into path events.

The development of the fault tree, thus far, has been a restatement of each event to increasing levels of resolution: from system, to subsystems, to paths. The top logic for the fault tree has been established. Next is to enumerate all the component fault modes, as well as the fault modes of support systems, that may interface with the individual path components. The top logic and the interfacing system events generally determine the degree of redundancy inherent in a particular safety system function. This is not always true, however, and the fault tree should be developed into the interfacing systems and into the control and power circuits to identify the more subtle, but important, contributions to risk. Also, some component fault modes will appear in more that one path, thus reducing redundancy for that particular fault mode. For example, rupture of any pipe downstream of the pumps and upstream of the injection valves (see Figure 1) will appear as a fault in the development for each path. This is to say, when the fault tree is converted to its simplest Boolean form the pipe rupture event will be a single fault. Knowing this, the top fault tree logic could be changed to reflect pipe rupture as a single event.

Figure 5 shows the conventional method for enumerating component fault modes and interfacing events. Each of the events shown within a circle is a basic component failure for which failure rate data are expected to be available. Events within diamonds are basic events that are not expanded because the event is judged not to be important, insufficient information is available, or the analyst merely wishes to postpone development. In any case, the event is given a name (see Section 5 below) and is accountable in the Boolean expression for the fault tree. The events within rectangles are interface events that will be expanded during the course of evaluating the interfacing systems (not evaluated herein).

The fault tree is developed as discussed until all components are identified in their basic fault states. The result is a binary model of the system that can be reduced to its simplest Boolean form. Failure rates, human error rates, and time intervals can be assigned to determine probability values for the components, subsystems, and system. Quantification involves naming of events and transferring all information contained on the fault tree to event tables and coding sheets for ease in the assignment of data to events and for computer processing.

3.2 Abbreviated Fault Tree Construction

Since all basic fault event statements on the conventional fault tree are subsequently transferred to tables, one way to reduce the analysis effort is to not put those statements on the fault tree. The first step in the abbreviated method, then,

is to enter all basic fault statements directly into fault summary tables (a portion of a fault summary table is shown in Table 1). Only the event code name, described in Section 5, is shown on the fault tree.

The second step is to define a new logic gate, the tabulation OR gate (see Section 4), to facilitate the listing of event names on the tree rather than to show named individual event statements within event type symbols as is conventionally done. Typically, systems that are evaluated contain a large number of events that are logically in series when reduced. For example, consider the fault tree development for two injection path components connected in series (see Figure 5). This development can be restructured (see Figure 6) where the code names for basic input events are listed under a tabulation OR gate: inputs to a component can be shown under the tabulation OR; otherwise, they can be expanded into their respective causes. Any number of components logically in series can be treated the same.

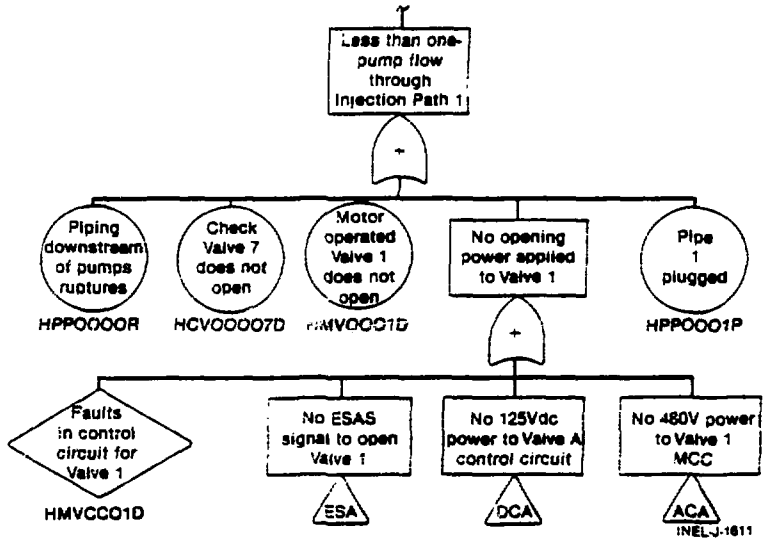

Figure 5. Enumerating component fault modes and interfacing events on conventional fault tree.

TABLE I. FAULT SUMMARY

Event Name	Event/Component	Failure Mode	Failure Rate	Fault Duration	Error Factor
P1P000RU	Pipe downstream of pumps	Rupture			
P1P011PL	Pipe 1	Plugged			
VCK071NO	Check Valve 7	Does not open			

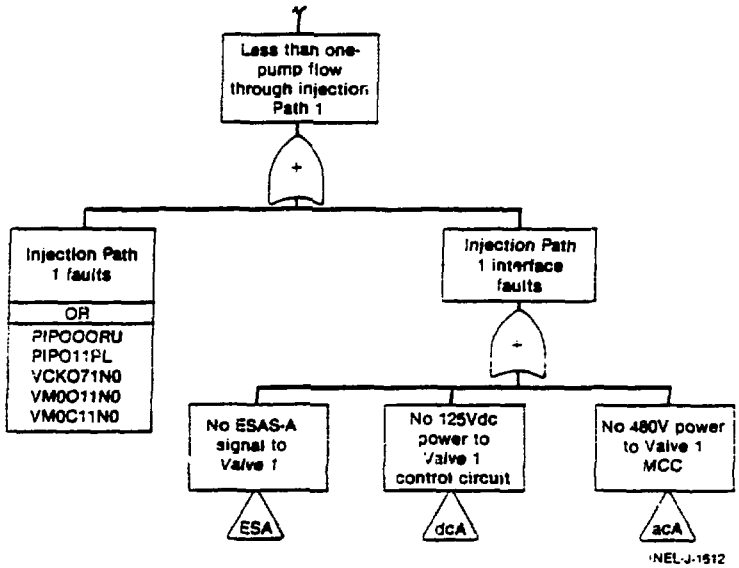

Figure 6. Basic fault events shown by code name only.

A completed tree would typically depict a top undesired event, basic fault events listed by code name under one or more tabulation OR gates, a few input events (to chains of components and to the system) identified within rectangles, a few house events, and the logic AND and OR gates used to relate the events. All other information is contained in the fault summary table.

4. GATE TYPES

A number of variations of the basic AND and OR gate types used to handle special situations have been introduced in the literature. Figure 7 shows the standard AND and OR gates and two other gates used in Browns Ferry Trees. The tabulation OR gate is used to enumerate a set of fault events that are associated with a series arrangement of components. Safety systems are typically comprised of redundant subsystems each having numerous components connected in series. A conventional fault tree constructed for one of these systems will have, then, a large number of OR gates, each with several inputs. The advantage of the tabulation OR gate is that it permits all the fault events for a series of components to be tabulated rather than spread out, sometimes over several pages, within individual fault symbols connected together by OR gates.

The combination gate simplifies the task of showing several combination of subsystem events, each containing the same elements (faults). For example, the high-pressure injection system shown in Figure 1 may require that two of the three pumps operate for a particular reactor coolant system break size. Also, numerous control systems incorporate coincident logic, such as two-of-four taken twice or two-of-three. In evaluating these system, it is necessary that the combinational fault logic be reflected on the tree.

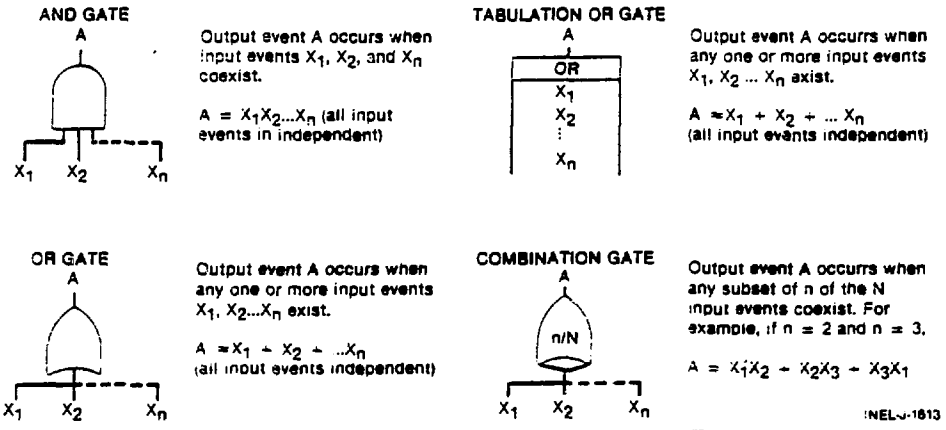

Figure 7. Abbreviated fault tree logic gates.

5. EVENT NAMING

In order to facilitate computer handling of events and to simplify fault tree construction, each nonexpanded event on the tree is given a code name to uniquely identify the event. The event may appear on the tree in several places, in each instance given the same name. An eight-character code was used on the Browns Ferry fault trees, identifying the event by system, type of component, component identification, and mode of failure. Human error events are included by virtue of the component affected and the component fault mode used.

6. REQUIRED CONDITIONS

A system can assume a variety of possible off, standby, or normal operational states, depending on plant conditions and operational requirements. For example, a water pump may be off if the water level in a tank is high but on if the water level is low; a diesel generator may be required to start if the offsite power fails; or a valve may be required to close if a fault has occurred in a downstream component. In fault modeling, inclusion of the conditions upon which a system or component is required in the analysis is important. A system fault is not considered a fault unless the system is required. For example, failure of a diesel to start at any time other than when the diesel is needed is not a fault insofar as the analysis is concerned.

Required conditions in a fault tree analysis can be in the form of explicit assumptions and the fault tree constructed accordingly, or the required conditions can be incorporated directly in the fault model. The latter is preferred because it provides versatility in the use of the model. When incorporated into the model, required conditions are shown within the "house" symbol. The "house" serves as a switch to turn on those events that are faults when the required conditions exist, and off when the required conditions do not exist. The "house" is usually input into one input of an AND gate, and the subtree of faults is input into other inputs of the AND gate as shown in Figure 2.

The house is also used to describe mutually exclusive faults, in which case, two "houses," are used—one or the other house can be on but not both at the same time. The house is also frequently used to classify faults for which each fault classification results in a different consequence. For example, in the evaluation of a reactor containment, classification of breach areas (faults) according to size may be desirable.

Any other condition pertinent to the analysis and affecting the analyst's thinking about the evaluation should also be specified. For example, knowing that a large LOCA has occurred and that suddenly large loads are to be placed on the electrical system should guide analysis of the electrical system.

7. HUMAN ERRORS

Human errors are relatively high probability events; therefore, human intervention or human inputs to components are important contributions to the probability of system failure. Switches, valves, adjustment pots, and test plugs are only a few of the many components that are subject to normal human input. All potential human errors are identified on the fault tree at the component where the human intervention takes place. For example, if a valve can be operated only from a switch in the control room, the human error event is associated with the switch in the control room, not the valve. If the valve can be operated remotely and locally, then the human error fault events should appear both places. Human errors are shown on the tree and in the fault summary as a mode of failure for the particular component subject to the human error.

8. TEST AND MAINTENANCE

System outages due to tests and maintenance and the human errors that can accompany these activities can be important contributors to the risks of nuclear plants, though some systems and components are tested and maintenance performed when the reactor is shut down, and therefore are not an important risk factor. When on-line testing and maintenance are required, a system that is redundant can become nonredundant during performance of the tests and maintenance unless override features have also been provided.

Outage due to test or maintenance is shown on the abbreviated fault model by an additional component fault event on the fault tree and on the fault summary for any subsystem or portion thereof that is unavailable during test and maintenance. Although not a failure in the strict sense of the word, outage is treated as a basic component fault with a mode designation "test" or "maintenance," and is so designated in the fault mode code. Unless each component is tested or maintained separately and at different times, only the component requiring the longest outage time is shown as a fault time. If each component is tested or maintained separately and at different times, each component is treated as a test and maintenance fault.

If a valve or other component can be left in the wrong state as a result of a test or maintenance error, the fault is also shown on the fault tree, and is treated as a human error as discussed in Section 7.

9. ANALYSIS STAGING

The abbreviated fault tree analysis helps stage the effort. That is, the analyst can determine the overall logic of complex and multiple systems before performing a detailed examination of components, thus, allowing the analyst to identify the more important, or critical, paths of the system without wasting time on details that may in the end be unimportant. The analyst constructs the abbreviated fault tree without identifying the individual events normally listed under the tabulation OR gate.

Instead, each tabulation OR is treated as a single component until the fault tree is reduced to its nonredundant Boolean form. Then, only those tabulation OR gates that appear as critical cut sets in the nonredundant Boolean form are expanded to include individual component events.

However, caution should be exercised: first, the tabulation OR gates must be independent of each other (they should not contain common elements if expanded), and second, reliance on the importance of tabulation OR gates resulting from staging can ignore potentially significant common cause events among those individual component fault modes not included.

Analyses of the Interim Reliability Evaluation Program were staged according to the "parent tree, daughter tree" concept,[7] where the daughter tree describes the enumerated individual component faults under tabulation OR gates, and the parent tree describes everything else, i.e., the top fault events, the interface events, the tabulation OR outputs as individual events, and the logic gates that relate those events. The parent tree is constructed first, the daughter tree being deferred until assessment is made of its need.

REFERENCES

1. "Reactor Safety Study," WASH-1400, NUREG-75/014, (1975).

2. N. H. Roberts et al., "Fault Tree Handbook," NUREG-0492, (1981).

3. R. H. Jennings et al., "Test Reactor Risk Assessment Methodology," ANCR-1271, (1976).

4. "CRBRP Safety Study, An Assessment of Accident Risks in the CRBRP," CRBRP-1, (1977).

5. Consumers Power Company, "Probabilistic Risk Assessment Big Rock Point Plant," (1981).

6. Injection lines may be designed for high impedance (small size) when more than one line is required to produce sufficient flow. If so, the logic would change reflecting less redundancy.

7. NRC Division of Systems and Reliability Research, "Interim Reliability Evaluation Program Phase II, Procedure and Schedule Guide," Draft Revision 2, (1980).