[redacted]

A Generic Task Approach to a Real Time Nuclear Power Plant Fault Diagnosis and Advisory System

Presented at

International Workshop on Artificial Intelligence for Industrial Applications

Hitachi City, Japan

May 25 - 27, 1988

AC02-86NE37965

CONF-8805230--1

DE89 003405

Presented at the International Workshop on Artificial Intelligence for Industrial Applications, Hitachi City, Japan May 25 - 27, 1988

A GENERIC TASK APPROACH TO A REAL TIME NUCLEAR POWER PLANT FAULT DIAGNOSIS AND ADVISORY SYSTEM

by

B. K. Hajek, D. W. Miller
R. Bhatnagar, J. E. Stasenko, W. F. Punch III
N. Yamada*

The Ohio State University
Nuclear Engineering Program
1133 Robinson Laboratory
206 West 18th Avenue
Columbus, OH 43220

*Hitachi, Limited
Hitachi, Japan

Abstract

A generic task toolkit developed at The Ohio State University Laboratory for Artificial Intelligence Research (LAIR) has been used in the development of an aid for operators of nuclear power plants. The toolkit consists of high level programming tools that enable knowledge to be used in accordance with its need. That is, if diagnosis is the need, a framework for performing diagnosis is provided. The operator aid provides for monitoring the conditions in the plant, detecting abnormal events, and providing the operator with guidance and advice through procedures on what path should be followed to mitigate the consequences.

Introduction

An operator aid to monitor the status of an operating nuclear power plant is under development at The Ohio State University. This system will monitor plant status, validate sensor data, diagnose plant faults, and provide procedure management for the operator. While the operator is following the procedure, the expert system will monitor the operator's performance and the plant's response to various operator actions, and will provide backup procedural steps for those that fail.

Reactor operators are presented with an overwhelming array of plant parameters and system statuses to monitor and interpret. They are assisted in this task by many sensors throughout the plant that provide readings in the control room such as flow rates, temperatures, pressures, power levels, valve positions, and motor operating conditions. Many of these sensors also have alarms associated with them.

Normal operating practice requires the operator to be involved in a number of activities that preclude his attention to all the information provided to him at any one time. Therefore, detection of any abnormal condition usually doesn't occur until one of the many alarms in the control room activates.

Once an alarm sounds, experienced operators often respond intuitively, having experienced the condition before either on the plant or during their simulator training. A second crew member will confirm the actions taken by reference to an alarm or abnormal procedure usually kept in one of many notebooks in the control room. For more severe conditions, the current practice is to follow procedures known as Emergency Plan Guidelines (EPGs) or Emergency Operating Procedures (EOPs).

Thus, an operator's job can be divided into three components:

1. Monitoring plant conditions,

2. Diagnosing a detected fault, and

3. Taking procedural action.

Three expert systems are under development to assist the operator in his performance of these three components. These three expert systems are tied together in the context of the system architecture shown in Figure 1.

The Plant Status and Monitoring System (PSMS) [1] provides the primary link with the power plant. It continually monitors the contents of a database looking for system changes that exceed preprogrammed limits. These changes may meet the entry conditions for various plant fault recovery procedures, or may indicate that conditions are not as expected and should be investigated. If investigation is required, a Diagnosis and Sensor Validation System (DVS) will be activated. [2] If a procedure entry condition has been established, a Dynamic Procedure Management System (DPMS) will be activated.

If DPMS is activated, DVS will operate in parallel to validate the data used in the diagnosis, and to provide further diagnosis. If DVS is activated, it can provide a diagnostic conclusion to DPMS. This conclusion will provide verification that the procedure initiated by DPMS should continue, or possibly a second procedure to be run in parallel can be retrieved or formulated to correct the plant malfunction, or to maintain the plant in a safe condition by maintaining the safety goals.

The entire system is being developed on Xerox D machines using InterLisp D, LOOPS, and generic task tools which run on top of these languages. (The tools have been rewritten to run on KEE, and are currently being rewritten to run on Common LISP. An extension of our work will have our system running on KEE in the near future.)

The intelligent database has been written in LOOPS, as has PSMS.

DVS has been written using the generic task tool known as the Conceptual Structures Representation Language (CSRL) which is a tool for hierarchical classification and diagnosis. [3]

DPMS has been written using a subset of the Design Specialists and Plans Language (DSPL), a generic task tool for object synthesis and refinement. [4]

Generic Tasks

Research at LAIR has centered on the concept of the generic task. [5,6] The premise of this work is that many ordinary cognitive processes are one of, or composed from, a small set of basic tasks, such as classification or planning. Each task is tuned to do its job by combining the separate elements of data structures (qualitative or quantitative data, and the relationships among the data - that is, the domain knowledge) with the method or inference strategy associated with the use of that data for the particular task.

For example, when a diagnostic problem is encountered, one problem solving strategy might be used. When a planning problem is encountered, another strategy will be used. The selected control or inference strategy is then applied to the domain knowledge.

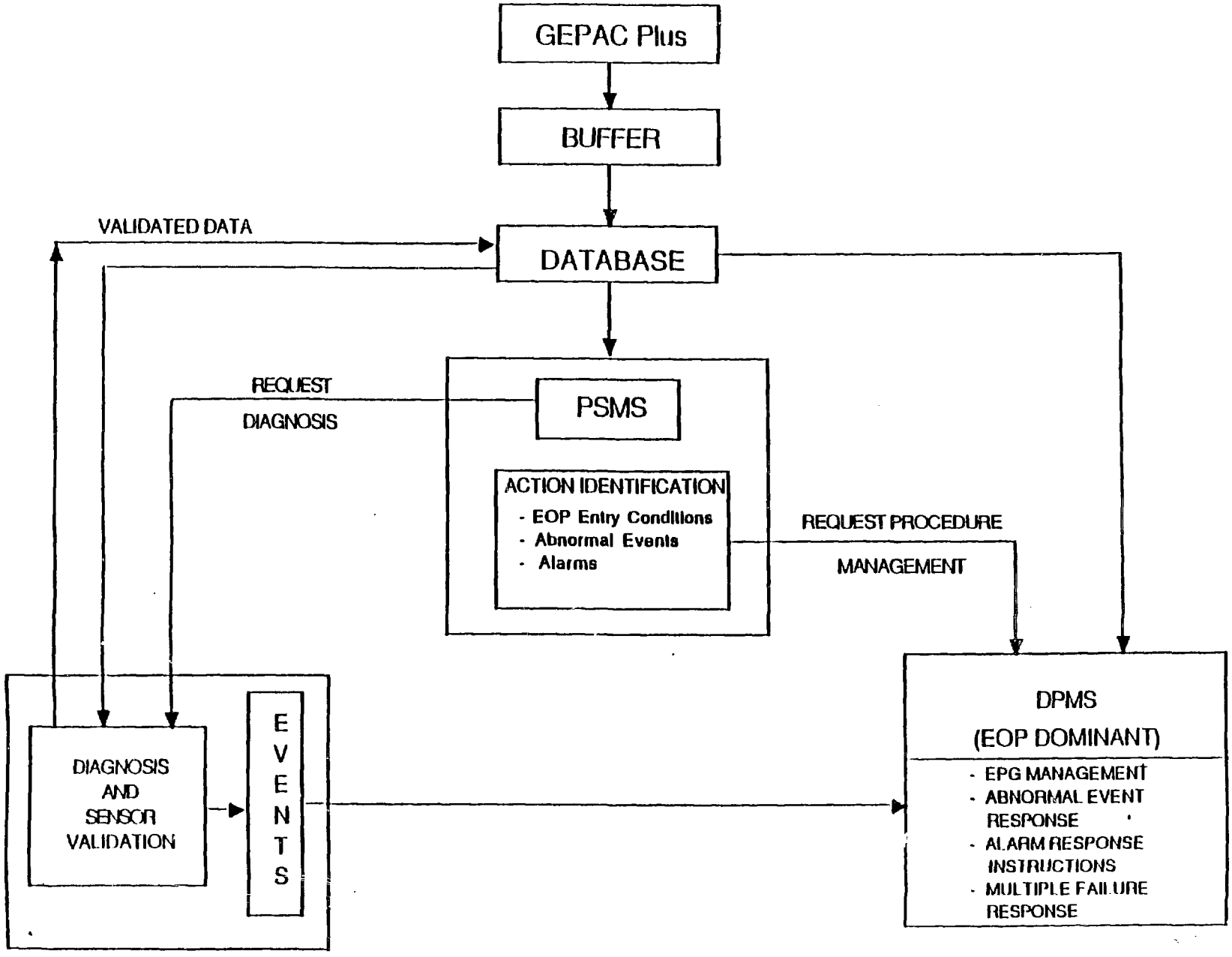
The result is that in the performance of a knowledge dependent task, the representation of knowledge cannot be separated from how the knowledge is used. In this sense, applying a specific inference strategy to domain knowledge may be referred to as a generic task. When domain knowledge is encoded with the inference strategy that comes with a task, we have a generic task problem solver. [3]

To assist the knowledge engineer in programming different tasks, a number of tools have been developed at LAIR. The expert system being developed at The Ohio State University takes advantage of two of these generic task tools.

The task of diagnosing faults in a system is a task that can be thought of as going from partial malfunction hypothesis, that is, determining that a malfunction is present, to a specific malfunction. To accomplish this, potential malfunctions are arranged in a hierarchical structure that uses an "establish-refine" control strategy. [3] For this task, we are using the generic task tool known as CSRL.

The task of selecting or forming a procedure, and following the performance of that procedure may be controlled by a planning system. [4] For this task, we are using the generic task tool known as DSPL.

Overall Structure of Integrated System



CSRL

Diagnostic problem solving may be viewed as a task of explaining a set of observations in terms of malfunctions that may have caused the observations.

In CSRL, a knowledge engineer produces a malfunction hierarchy which has more general classes of malfunctions at its higher nodes, and more specific malfunctions at the lower or successor nodes.

The establish-refine control strategy operates by evaluating malfunction hypotheses at each of the higher level nodes. If a particular node, or specialist, establishes (that is, the knowledge in the node indicates the fault is likely) the lower level, or daughter, nodes are considered. If a malfunction hypothesis (node) does not establish, it is ignored along with its daughter nodes.

This method of establish-refine enables the system to rapidly prune the hierarchy of malfunctions to quickly establish the specific malfunction that has caused the initial observations.

DSPL

Routine design, planning, or procedure following may be viewed as tasks having a common element of making well defined and appropriate choices in a domain.

Several types of knowledge are inherent in these choices: (1) knowledge of decomposing the overall plan problem into smaller, more manageable plans; (2) knowledge for ordering the execution of the sub plans; (3) knowledge for ordering the execution of these sub plans or procedural steps; (4) knowledge of appropriate constraint testing which helps to focus the plan to more quickly achieve the objective; and (5) knowledge to invoke backup plans or procedures when procedural constraints (such as equipment failures) are not satisfied.

DSPL supports the generation of various programming "agents" (subroutines essentially found at nodes) to appropriately use and invoke these different knowledge types.

In its basic operation, DSPL invokes a planning agent which attempts to fill in its plan elements. It does so by invoking a number of sub agents, each responsible for a portion of the plan. Upon completion, the plan is checked for consistency by evaluating constraints within the planning agent. If the plan is consistent, the job is done. If, however, inconsistencies are found, the plan is reformed taking into account the inconsistencies.

In the nuclear power plant domain, it is necessary to check the success of every step in a procedure. This check and failure handling when success is not achieved must be done in real time. The existing DSPL, as provided by LAIR, has been enhanced for our application to provide real time plan evaluation and failure recovery before completion of the plan.

Components of the Expert System

The expert system for providing operator assistance consists of the four main components shown in Figure 1 and previously discussed in the Introduction. Each component will be discussed in more detail in this section.

Intelligent Database

The Database for the expert system will receive its data from the plant process computer and from the plant Safety Parameter Display System (SPDS). It also will infer data that normally is not available to the plant operator.

The present design assumes the availability of the General Electric GEPAC Plus plant process computer replacement system. GEPAC Plus also contains a fully functional SPDS, thus simplifying the interface to the plant.

Using data directly from GEPAC Plus provides two main advantages. The first is that much of the raw data will have already been treated by an averaging process that will significantly reduce the number of points to be considered by the expert system.

The second advantage is that the data also will have been validated using routine data validation techniques. This assures a certain degree of confidence in the data, thus allowing the expert system to have the operator take action through DPMS without first performing its own data validation function. Nevertheless, DVS will operate in parallel with DPMS to validate the data used during the diagnosis by using context sensitive techniques.

The database serves all three sub-systems. It contains all the data available from the process computer, and techniques to answer higher level questions about various plant states based on available data.

In the database, data is organized in three classes based on the kinds of questions that can be asked about the data and the techniques required to answer these questions. The three data classes are:

1. Continuous or Analog Data. Questions in this class concern trends, values relative to normality, or length of time a value has exceeded a limit.

2. Component/System States. Questions in this class concern operational modes, component states such as tripped, open, closed, or operating, and system status or availability.

3. Bistable Component States. Questions in this class can concern alarm statuses or light indications, and only can be answered as ON or OFF.

PSMS

The database is continually monitored by the Plant Status and Monitoring System [1] which sends information to the diagnosis or procedure management systems whenever an initiating event (abnormal state) is detected.

Once an abnormal state is detected, PSMS performs the first level of decision making by initiating either DVS or DPMS. This decision is based on the safety goal hierarchy that establishes EPG actions, the abnormal events classification of the plant, the alarm response procedures, or preset sensitivity levels maintained in the database.

For any malfunction, PSMS operation must result in a response to:

1. Direct indications of plant changes that can be resolved by simple actions,

2. Changes in plant status where parameters are outside of normal ranges, but alarms or conditions do not satisfy entry conditions to one of the alarm response instructions or an event procedure, or

3. Multiple alarms/sensors indicating that a safety goal is being threatened.

For PSMS to operate in these three modes, it must have several properties.

It must be able to distinguish between normal and abnormal plant states. It also must be able to determine the status of the components and systems required to keep the plant operating in a safe state for the given operating mode.

It should be able to distinguish between normal and abnormal transients. Thus, it will need a knowledge of the normal operating parameters as a function of power and operating mode. It also will need to detect manual operator actions.

Finally, PSMS should be able to output messages to DPMS, DVS, and plant operating personnel.

DVS

When activated by PSMS, DVS will look for malfunctions by matching expectations in a malfunction hierarchy. If a malfunction node establishes, DVS will attempt to refine the diagnosis to find the root cause at the lowest available component level.

If a data point is found to be questionable, the value will be changed in the database according to values in an expectation pattern, and the hierarchy will be run again. It will iterate in this manner until a conclusion is reached and all data is found to be correct. At this point, it will provide DPMS either with confirmation that a proper procedure is being run, or with a recommendation that a different procedure should be followed.

As an example of how this system will operate, consider the case of a decreasing water level in the reactor pressure vessel. The specific malfunction is caused by a feedwater recirculation valve inadvertently opening with a concurrent failure of the feedwater level controller calling for no change in feedwater flow rate. To demonstrate how the sensor validation function operates, we also will assume the flow sensor in the recirculation line fails. The malfunction hierarchy for this failure is shown in Figure 2.

CSRL tests malfunction hypotheses in the malfunction hierarchy by first examining the most general nodes, located at the left of the tree, and then moving through the tree at the next lower level from the bottom up. Therefore, it is important to construct the tree with the nodes you want to have considered first at the bottom at each level.

A coolant system fault is detected because of the lowering reactor water level. LOCA does not establish because the required expectations, such as increasing drywell pressure (among others), are not present. However, the condition of Reactor Inventory Change does establish. This can be seen in the Confidence Browser in Figure 3 where a number 3 indicates high confidence (established), and a -3 indicates low confidence (rejection). Knowing the malfunction, one can surmise how each additional node is either established or rejected.

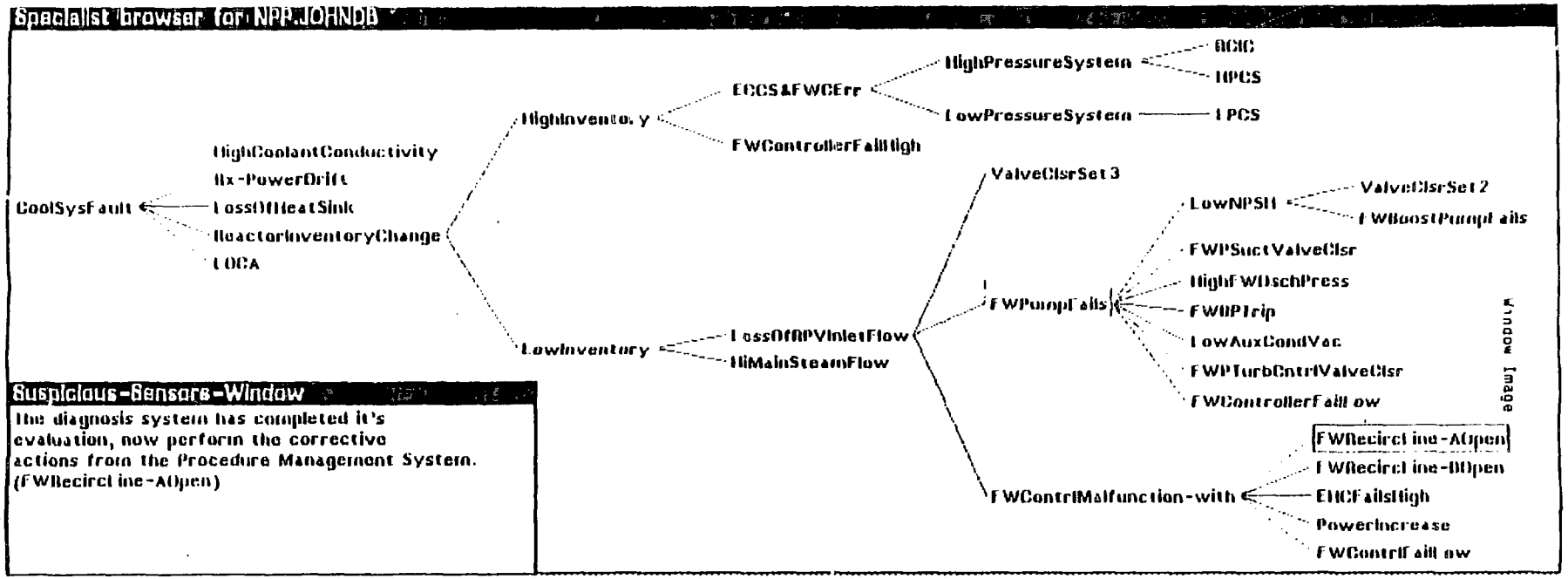


Figure 2. DVS Specialist Browser for Reactor Inventory Change Malfunction Branch.

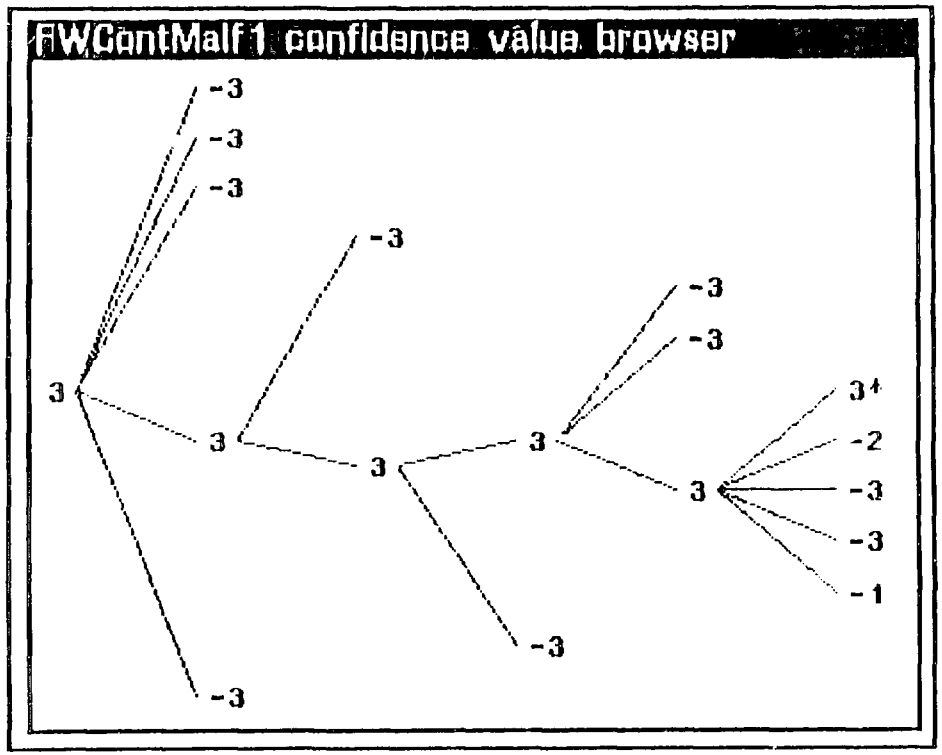


Figure 3. Confidence Browser for a Malfunction With Questionable Sensor Data.

window image

When the given malfunction establishes, it does so with a 3*. The * indicates a possible sensor malfunction. By examining the Knowledge Group for this malfunction in Figure 4, it can be seen that either the flow rate sensor or the hot surge tank level sensor can cause the *. Because the table is reviewed from the top down, the value for the flow rate is changed in the database, and the hierarchy is run again.

For this case, the result is the same, but without the questionable sensor indication, as shown in Figure 5.

DPMS

DPMS operates in a safety function [7,8] maintenance mode.

The safety functions of the plant are organized in a hierarchical manner. This representation allows for failure handling, and can assure that the important safety functions are maintained in preference to taking actions for specific events that may or may not be identified.

The plans within DPMS include the event oriented abnormal procedures, alarm response procedures, and the symptom oriented EPGs. These procedures are integrated through the safety function hierarchy. The EPGs are the upper level functions of this hierarchy, while the event procedures fill the lower nodes.

Entry into the EPGs can occur in either of two ways: (1) By direct initiation from PSMS upon the occurrence of an entry condition, or (2) By failure of a lower level node procedure resulting in further plant degradation.

Any time one of the entry conditions to the Emergency Procedure Guidelines is achieved, DPMS will assure that these procedures are followed. Thus, the system is EPG dominant. While the EPGs are being followed, DVS will try to determine a specific fault, and if possible, will provide DPMS with information that will allow a secondary procedure to be run in parallel with the EPGs so the consequences of a malfunction might be mitigated sooner.

DPMS will receive instructions from PSMS as to which procedure is to be followed. It may, as previously stated, also receive instructions from DVS to (1) confirm appropriate actions, or (2) follow an alternate procedure, or (3) follow an additional procedure along with the current procedure.

During procedure performance, each step will be monitored to assure its success. If a step is unsuccessful, a backup procedure will be provided for that step. Likewise, if a step is expected to fail due to the unavailability of equipment, a backup procedure will be provided.

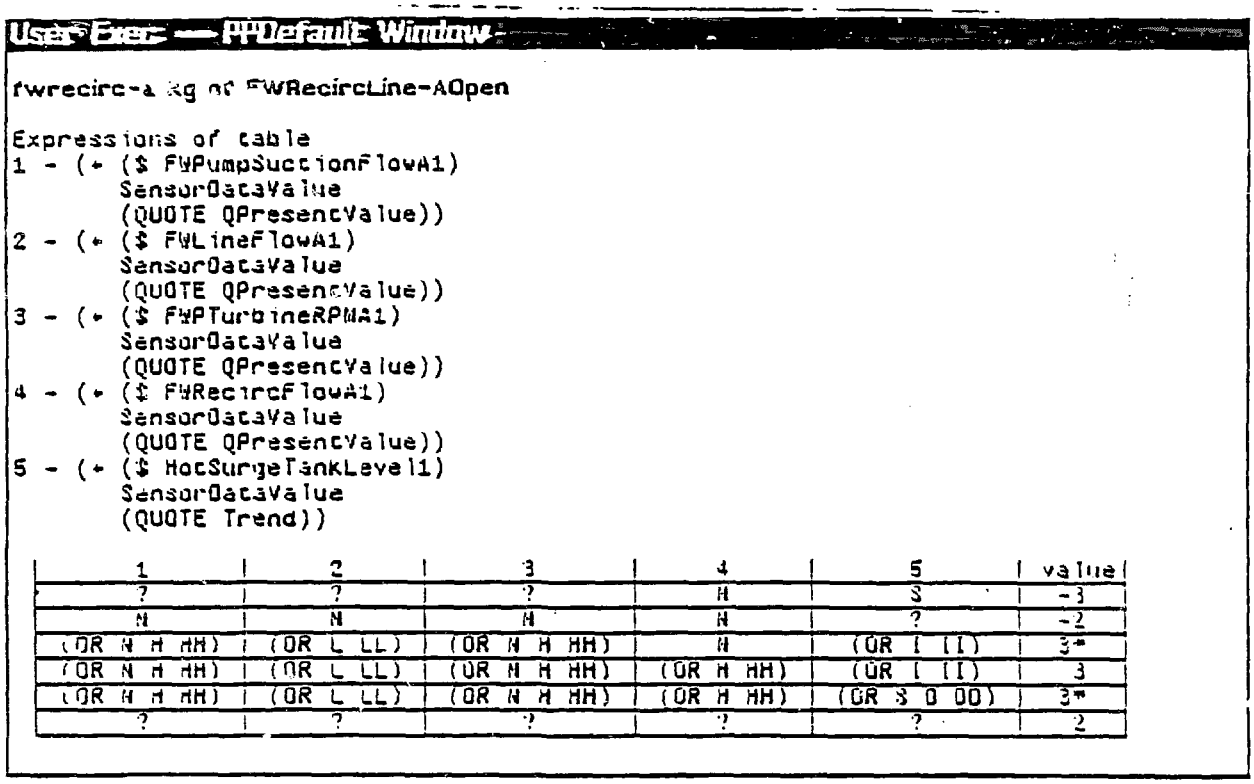


Figure 4. Knowledge Group for Feedwater Recirc Line A Failed Open.

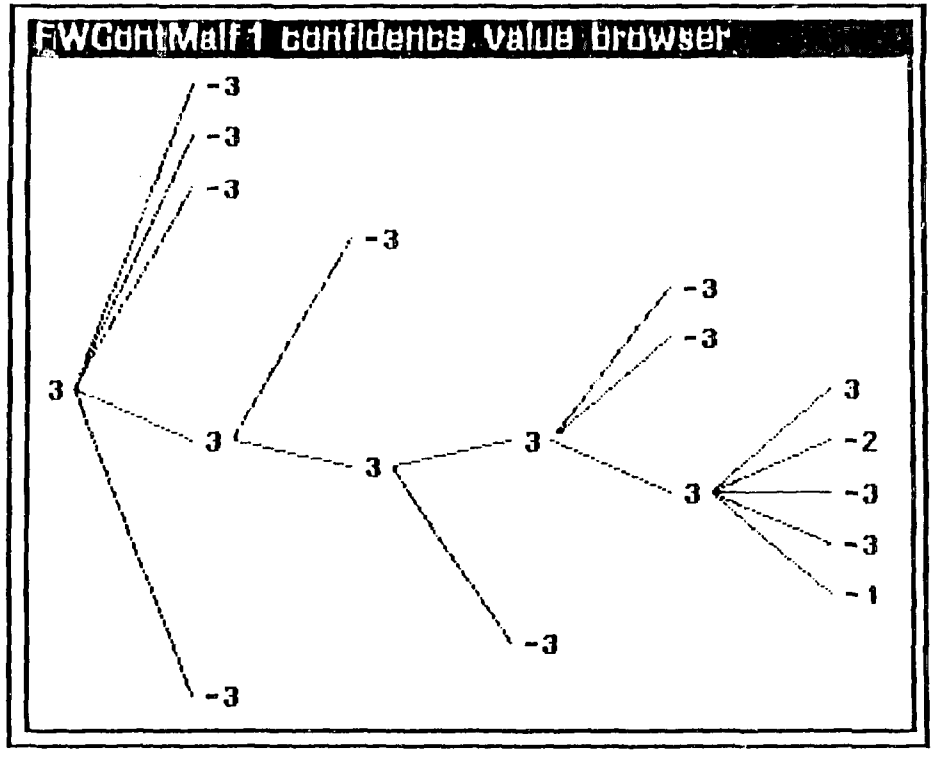


Figure 5. Confidence Browser for a Malfunction After Questionable Sensor Data Has Been Changed by DVS.

Window Image

System Testing

The development of the expert system has used the Perry Nuclear Power Plant, a General Electric BWR-6 design, as the reference facility. Plant statuses and operating conditions for Perry have been programmed into PSMS and DVS. Perry procedures have been used for the bases of procedural actions to be specified by DPMS.

The expert system is being tested by using the Perry plant referenced simulator. Transients are run on the simulator while data is collected. Later, this data is reviewed for indicators that define the transient of interest. This data is programmed into PSMS or DVS to provide expectations for each node in the malfunction hierarchy. The expert system is run to refine the expectations and to develop tables to detect ambiguous sensor data.

Summary

Using the two generic task tools, CSRL and DSPL, design work on DVS and DPMS is nearing completion. The work on PSMS in LOOPS is scheduled for demonstration by April 1, 1988. A demonstration of the entire system (all three sub systems integrated as one) will be available at that time.

DVS has already been demonstrated to be effective in diagnosing faults and validating sensor data in a BWR coolant system. DPMS has been tested for a reactor scram procedure in a previous form, and still needs to be tested in its present version. Testing of the complete system will be demonstrated for faults in the condensate-feedwater system.

Acknowledgments

The authors would like to acknowledge the contributions of B. Chandrasekaran, S. Hashemi, and D. D. Sharma.

The development of the two prototype systems was partially supported through a grant from the National Science Foundation (Grant No. 8400840). The research described in this paper is supported by The U. S. Department of Energy (Grant No. DE-AC02-86NE37965).

We also would like to express our appreciation to the staff of the Perry Nuclear Power Plant Nuclear Training Department for their extensive assistance in obtaining plant data and for working with us to run plant transients on the Perry simulator. This collaboration has provided a reference plant for the work we are performing.

The collaboration of GE Nuclear Energy, Electronic and Computer Services, also is appreciated. GE's collaboration is making it possible to assure that our expert system will properly interface with the plant computer systems.

Finally, the authors wish to acknowledge the support provided by The Ohio State University.

References

1. Hajek, B. K., Stasenko, J. E., Hashemi, S., Bhatnagar, R., Punch III, W. F., and Yamada, N., "The Structure of an Expert System to Diagnose and Supply a Corrective Procedure for Nuclear Power Plant Malfunctions," presented and published in the Proceedings of the ANS Conference on Artificial Intelligence and other Innovative Computer Applications in the Nuclear Industry, Snowbird, Utah, 1987.

2. Hashemi, S., Hajek, B. K., Miller, D. W., Chandrasekaran, B., Punch III, W. F., "An Expert System for Sensor Data Validation and Malfunction Detection." presented and published in the Proceedings of the ANS Conference on Artificial Intelligence and other Innovative Computer Applications in the Nuclear Industry, Snowbird, Utah, 1987.

3. Bylander, T. and Mittal, S., "CSRL:A Language for Classificatory Problem Solving and Uncertainty Handling," AI Magazine 7(2):66-77, 1986.

4. Brown, D. C., and Chandrasekaran, B., "Knowledge and Control for Mechanical Design Expert System," IEEE Computer 19(7):92-100, 1986.

5. Chandrasekaran, B., "Towards a Functional Architecture for Intelligence Based on Generic Information Processing Tasks," presented and published in the Proceedings of the Tenth International Joint Conference on Artificial Intelligence, Milan, Italy, August, 1987.

6. Chandrasekaran, B., "Generic Tasks in Knowledge-Based Reasoning: High Level Building Blocks for Expert System Design," IEEE Expert, Fall, 1986, pp. 23 - 30.

7. Sharma, D. D., Miller, D. W., and Chandrasekaran, B., "Design of an Artificial Intelligence System for Safety Function Maintenance," Transactions of the American Nuclear Society, Vol. 50, pp. 294 - 297, 1985.

8. Corcoran, W. R., et. al., "The Critical Safety Functions and Plant Operation," Nuclear Technology, Vol. 55, pp. 690 - 712, 1981.