SRC TR 88 - 38

GOTRES: AN EXPERT SYSTEM for FAULT DETECTION and ANALYSIS

by

Donald T. Chung
and
Mohammad Modarres

GOTRES: AN EXPERT SYSTEM FOR FAULT DETECTION AND ANALYSIS

Donald T. Chung

Mohammad Modarres

Department of Chemical and Nuclear Engineering

The University of Maryland

College Park, Maryland 20742

R. Niall M. Hunt

Baltimore Gas and Electric Company

Calvert Cliffs Nuclear Power Plant

Lusby, Maryland 20657

GOTRES: AN EXPERT SYSTEM FOR FAULT DETECTION AND ANALYSIS

Donald T. Chung
Mohammad Modarres

Department of Chemical and Nuclear Engineering
The University of Maryland
College Park, Maryland 20742

R. Niall M. Hunt
Baltimore Gas and Electric Company
Calvert Cliffs Nuclear Power Plant
Lusby, Maryland 20657

ABSTRACT

This paper describes a deep-knowledge expert system shell for diagnosing faults in process operations. The expert program shell is called GOTRES (GOal TRee Expert System) and uses a goal tree-success tree deep-knowledge structure to model its knowledge-base. To demonstrate GOTRES, we have built an on-line

fault diagnosis expert system for an experimental nuclear reactor facility using this shell. The expert system is capable of diagnosing fault conditions using system goal tree as well as utilizing accumulated operating knowledge to predict plant causal and temporal behaviors. The GOTRES shell has also been used for root-cause detection and analysis in a nuclear plant.

1. INTRODUCTION

Recent applications of diagnostic expert systems for engineering problems have favored representing the underlying knowledge through the use of deep-knowledge methods. These methods use system descriptions to capture the knowledge required for complex diagnosis. The system descriptions are means of modeling the system and can take many forms. The selection of the appropriate model depends heavily on the nature of the problem. Of the models proposed, many have been based on system structure and fall into several categories. Typical of this approach are the models proposed by Genesereth, Davis, and Laffey (1-6) which have been demonstrated in electronic systems. These modeling methods rely on the serial nature of systems. The models capture system's parts and interconnections and rules to project the proper signal outputs from components based on signals entered. The general fault diagnostic approach used in these methods depends on the passage of electronic signals through the interconnected structure. Such structure-based models are successfully demonstrated in diagnostic applications in electronics in which the flow of electrical signals can be used to trace down component failures.

A second type of model, proposed by Milne (7), is based on the "theory of responsibilities." In his approach, responsibility for parts of the output is assigned to various parts of the system. Kramer and Palowitch (8) have proposed a method in which potential origins of failure are located by analyzing violations of system operational constraints. These two models tend to be less restricted by the problem type.

Rich and Venkatasubramanian (9) have proposed a fault tree deep-knowledge model which employs less limiting causal knowledge. However, direct diagnosis through a fault tree is event-oriented and thus overly detailed to perform on-line diagnosis. Ulerich and Powers (10) use the cut sets from a fault tree of a system to perform diagnosis. The diagnosis through these cut sets requires a less detailed search but does not exhibit intuitive and transparent reasoning. In on-line diagnostic applications, the transparency of the reasoning process will usually determine whether the user accepts the results.

This paper will present the use of the Goal Tree-Success Tree (GTST) method for modeling deep-knowledge for fault diagnosis

(11). In our implementation, the principles of GTST are slightly modified. The modification occurs at the end points of the goal tree, where we have used operating knowledge (in the form of if-then rules) instead of success tree. This modification make the model more suitable for on-line diagnosis. An application using this method is described in detail; other possible applications are also reported.

The GTST model has a tree-like hierarchial structure which represents the basic underlying principles of a complex system. Each node in the tree represents a goal. The lowest level goals are often supported by hardware and human activities. The overall structure is developed based on a problem reduction strategy. The lower level goals explain "how" an upper level goal is achieved, and an upper level goal shows "why" the lower level goals are needed.

An expert system shell designed to accept knowledge modeled in the GTST format has been developed and tested in a process plant environment. The diagnostic expert system has been implemented on a scaled-down model of the primary side of a typical Babcock & Wilcox commercial nuclear reactor. In this paper we have presented a description of the GTST knowledge-base, the problem-solving approach, and the on-line use of this expert system.

2. PROPOSED APPROACH

Goal tree-success tree is a structure which can be used to organize complex systems and their fundamental knowledge into a format suitable for problem-solving. The GTST structure is conceptually very simple and thus easy to use. At the same time, it is descriptively robust and can accommodate all aspects of system complexity, interactions and interdependence. In the power plant environment, the GTST will provide a top-down hierarchical model of goals and subgoals which are required to achieve a pre-defined plant objective. Specific rules for developing the GTST are employed to maintain hierarchy and completeness (12). The first step in the development of the tree requires identification of an unambiguous and definitive top goal or objective. It is from this definition that each discrete plant goal and subgoal necessary for the achievement of the stated objective can be identified.

Development of the goal tree progresses vertically downward from the objective, in levels, to progressively more detailed lower levels. At each level, the GTST developer dissociates the goals into a necessary and sufficient set of dependent subgoals. As the vertical development of the tree progresses, specific rules are applied at each level to ensure GTST accuracy and completeness, and also to ensure that the proper hierarchy between goals and subgoals is rigorously maintained. Important rules during the GTST development are,

- * When looking downward from any goal towards the bottom of the tree, it must be possible to define explicitly <u>how</u> the specific goal or subgoal is satisfied.
- * When looking upward from any subgoal towards the objective or top-goal, it must be possible to define explicitly <u>why</u> the specific goal or subgoal must be satisfied.

When the lower level goals reach a detail level where further dissociation is not possible without reference to hardware, the nature of the tree changes. From this point downward the tree represents the hardware requirement, and success trees can be used to represent the combination of components (i.e., a success path) that must work for the goals to be successful. Figure 1 shows a typical GTST structure.

Because, at all levels of the goal tree part of a GTST, each goal contributes in some measure to the success of the top objective, each goal must be related to all others with "AND" logic. This is not to mean, however, that all goals in the tree are equally important. In fact, individual goal importance is determined by the frequency with which it is expected to fail, and the impact which its failure has on the higher level goals and the objective.

It must be recognized that logically equivalent success and failure trees can always be constructed, so that goal tree analysis can proceed with either orientation. However, experience has shown that success orientation is preferred since it tends to lead to better and more succinct definition of process functional goals. What follows are discussions as to why this may be so.

When one is attempting to understand the qualitative unreliability characteristics of an industrial process, fault descriptions are more easily grasped than are success descriptions because the process is typically being viewed externally, from the perspective that a failure has already occurred. This means that the person is likely to be looking at the process in a deductive manner.

Suppose that one has decided to drive from Washington, D.C., to New York City. Prior to departure and during the trip, the driver tends to visualize the roads before him as a set of success paths in which one is planned and others are available, should an alternate path be required to circumnavigate obstacles encountered along the way. The driver does not think of the road ahead as a set of potential failures which prevent his arrival. Rather, he tries to determine problems along the way, then effect on his goal, and use all available means to achieve such a goal. This hypothetical situation closely resembles the mentality of an operator in a nuclear plant control room environment in which the operator wants to achieve the goals of safety and economy. In these cases the human behavior is forward-looking, or inductive in nature, and success orientation appears most consistent with human needs.

Suppose however, we view this trip from a different perspective. Namely, that of a person who is awaiting the arrival of our driver in New York. As the arrival time approaches and the car does not appear, the person waiting will start to think of all the things that could have gone wrong to prevent its arrival. Failure is presumed, and the reasoning for failure is deductive in nature.

The above scenario tends to lead to the conclusions that when a human is engaged in deductive reasoning processes, failure orientation appears to be preferred, whereas in an inductive situation, the human appears much more comfortable using success orientation.

3. EXPERT SYSTEM SHELL BASED ON GTST

An expert system shell called GOTRES (GOal TRee Expert System) in which the GTST type structure forms its knowledge-base has been developed. The GOTRES shell consists of interactive front-end functions which create the fundamental knowledge-base and a unique inference engine which searches the GTST structure.

The shell is written using Golden Common LISP, and runs on an IBM AT computer. Figure 2 illustrates the basic structure of the GOTRES program. Various elements of GOTRES shell as shown in this figure will be discussed in this section.

Goal verification is based on plant parameter values (called indicators). For each goal, specific regions of operation have been identified as successful using relevant indicators. These indicators are either plant instrument readings or some calculated parameters. Knowledge regarding indicators that should be used to identify regions of successful operation and the range of values for defining such regions are acquired through examining the plant operating principles and through interviewing experts.

Instrument readings and calculated parameters are the primary indicators used in diagnosing fault conditions. Calculated parameters are indications, such as "difference in temperature" or "rate of change in temperature," which are calculated using several instrument readings. If these indicators cannot positively indicate goal success or failure, then questions that query the operator for additional observations can also be used. In diagnostic operation where such application is intended, the questions have to first be predetermined and included in the knowledge-base. In our on-line application (discussed later in this paper) the diagnosis was based solely on the primary indicators.

The GTST model of the plant, the plant instruments and calculated parameters, together with the fault-free operating ranges are stored in the knowledge-base as frames (13). An example of a GTST frame is displayed in Figure 3. While each goal is represented by a frame, only one frame is used to represent all of the indicators and their fault-free operation ranges. Each goal frame contains a list of indicator names which provide the criteria for indicating its success. The instrument frame contains slots for all the parameters normally used in the diagnostic process. Instruments and calculated parameter names are installed in the instrument frames, and each indicator has an upper and a lower bound value as its attribute, which identifies its success region. The inference engine search process identifies plant faults by comparing measured or calculated plant parameters against the fault-free operating ranges, thus identifying failed goals, and traces the progression of failed goals down the GTST structure.

Although GOTRES is based on deep-knowledge modeled in GTST, causal and temporal behaviors that are observed through studying experimental or analytical results can also be implemented in the expert system. In our approach, GOTRES uses only the goal tree part of the GTST to inductively identify the possible cause of a fault condition so as to better formulate corrective or alternate methods of achieving the objective. Upon reaching the lowest level of the goal tree, through a given goal tree path, the previously observed system behaviors (when such a path existed) serve to supplement very specific information to the diagnostic process. For example, predicting expected system behavior or formulating potential corrective actions.

The observed system behaviors are established as causal and temporal rules that provide advisement to the operator. Causal rules are used to relay underlying cause and effect conditions and are activated when their indicators are in the success range. Temporal rules are similarly activated to provide advisement for expected behaviors. Sometimes, for a given temporal behavior, more than one temporal rule is used; each representing a range of values that success indicator can take.

In addition to causal rules which show observed cause and effect relationship, the specific hierarchy in each goal tree path provides a functional causal chain and thus likely plant functional behavior. That is the causal chain, which leads to the loss of each higher level goal and which can be easily traced when a path within the goal tree is diagnosed as the likely cause of the loss.

GOTRES currently does not identify specific components that have failed since instrument values and calculated parameters are not sufficient for conducting detail tests on the component level. To test specific components a success tree would be used, and the diagnostic process would require substantial use of additional indicators. For on-line diagnostic operation, this would require an operator to be available to observe plant components and to answer GOTRES queries. This can slow the effectiveness and performance of GOTRES and its on-line capabilities.

The problem-solving method in GOTRES is performed by tracing path(s) of failed goals. The inference engine uses depth-first search (14) starting from the top goal to find a trace of faults (ie. to find a fault path). Having identified a fault path at the lowest level of the GTST, the search process returns to each preceding level to check additional failed goals (in case that there are multiple paths). Depth-first search is reapplied if additional faults are detected. Figure 4 shows a typical search.

The control structure of the GOTRES shell for executing this search is shown in Figure 5. The CHECK function is an initiating point in the shell for testing goals. A function called TEST determines whether a goal is successful. If it is, then other goals on the same level are called by ON-SAME-LEVEL and passed to CHECK. If a goal has failed, then CHECK-END examines whether it is at the lowest level of the tree by checking for any subgoal. If a subgoal is found, it is passed to CHECK and the process is repeated to reach and find the lowest level failed goal(s). When the lowest level has been reached, and the fault path identified, then UP-CHECK function takes the search process to the preceding GTST level, and CHECK any remaining goals for additional fault path.

Failed paths in the GTST which lead down to the lowest goal level are recorded during the search process. Since GTST has a problem reduction structure, the list containing the description of goals in a failure path offers an intuitive explanation of the reasoning process. An explanation facility, based on failed goals, is also available in this expert system.

A depth-first search process is selected over others, such as breadth-search, because plant fault conditions are more frequently caused by single failures. Using depth-first search the inference engine is most efficient in treating single-failure fault conditions. Specifically, the maximum number of goals that should be tested in order to identify a single failure is proportional to the number of levels. Namely, if C is the average number of subgoals for each goal (including the top goal), and N is the number of levels, then the maximum number of goals which will be tested by depth-search is: C X N. In breadth-search, the maximum number is geometrically related to the number of levels. The maximum number of goals that will be tested by breadth-search is therefore C^N.

Typical GTST structures, on the average, have goals with greater than two subgoals. In structures of this type, depth-search is the most efficient for finding single failures. In systems which exhibit atypical structures such as the one shown in Figure 6, or systems prone to having a large number of multiple-failure fault conditions, a combination of breadth-search and depth-search, or only breadth-search method can be more efficient. Our current research is focused on studying the various types of GTSTs to formulate a strategy to optimize the search process. This strategy will be modeled as a set of so-called "meta-knowledge rules" for devising a proper search to be used by the inference engine when a new GTST structure is used.

4. SAMPLE APPLICATION

As noted earlier the GOTRES has three components:

1. Knowledge-base: This component forms the source of plant knowledge and must be prepared and entered by a user.
2. Inference engine: Currently, this component uses a fixed search routine which is a modified depth-search.
3. Instrument readings (indicators): This component provides plant parameters and conditions and forms the factual knowledge for diagnoses.

The use of GOTRES in a diagnostic application begins with development of the process GTST model and follows these steps:

1. Define a process goal for monitoring.
2. Develop a GTST model of the process showing how the goal is accomplished.
3. Identify temporal behaviors or hardware-level casual

behaviors, which experts have observed from experimental or analytical results for each goal tree path, which can be effectively used in the diagnostic process.

4. Identify the instruments (and thus available parameters) and calculated parameters that are needed for testing the success of goals and behaviors.
5. Identify the success range for each available and calculated parameter.
6. Define the calculated parameters as LISP functions.
7. Install the GTST and instruments in the knowledge-base using the front-end program.

4.1 Description Of The Plant

The University of Maryland 2x4 Loop is a 1/500 volume scaled model of a B&W type nuclear reactor primary system (15). The Loop consists of a reactor vessel with core barrel and annular downcomer, two hot legs, four cold legs, two once through steam generators, a pressurizer (Figure 7), and a cooling tower. Heat addition into the Loop is accomplished by means of 15 heater rods. The objective of the Loop facility is to study the thermal-hydraulic behavior of a multi-loop system during a small break Loss Of Coolant Accident (LOCA).

The main data acquisition system (DAS) for the loop is computer-based and utilizes a Micro-VAX II computer and a NEFF

System 620 analog data acquisition system. The Loop instrumentation connections to the DAS consist of 150 channels, of which there are 120 thermocouples and 30 high level inputs (differential pressure cells, flow meters, etc.) The location of the instruments is shown in Figure 8. Data collection and analysis software have been developed to the extent that, currently during a test, approximately 150 instrument channels can be sampled at a rate of 5 times per second. For the performance of our test, a subroutine has been written in the data collection software for sending the latest data (i.e. measured parameters), by interrupt, to the IBM AT computer in which GOTRES resides.

4.2 Selecting GOTRES Diagnostic Objective

In the past, the operating of the 2X4 LOOP for simulating nuclear reactor LOCA conditions has occasionally resulted in the over heating and rupturing of electrical heating rods which are located in the LOOP reactor core. Although this condition presents no physical hazard to operators of the LOOP, the difficulty in replacing the heaters does shut the LOOP facility down for an extended period of time and is costly.

All the experiments conducted at the LOOP are designed to simulate the various LOCA conditions. Every experiment of this type requires coolant to be leaked from the reactor while the heater is producing heat. Maintaining sufficient coolant level in the reactor core is considered critical for preventing rod overheating. Yet, the coolant level is not easily measured, and there can be unexpected plant fault conditions that affect the temperature of the heaters.

Coolant level in the core is calculated from three differential pressure cells which measure the pressure at the top, middle, and bottom of the vessel. The difference in the measured hydraulic pressure between the top and the middle of the vessel is used to define the upper core coolant level. Similarly, the pressure difference between the middle and the lower core is used to define the lower core coolant level. These coolant level measurements are based on hydraulic pressure difference between the vertical locations, and depend on the density of the coolant.

During a simulated LOCA, the coolant density changes as a result of changes in pressure. When a LOCA occurs, the LOOP pressure drops suddenly, and the coolant proceeds to flush into vapor. The changes in coolant density distort the coolant level measurements. The operator's responsibility during a simulation is to collect data that would be representative of a certain type of LOCA condition. The operator will not always be able to recognize the existence of unexpected additional faults that may affect the heater temperature as well as the data being collected.

4.3 Knowledge Acquisition

4.3.1 Goal Tree

Once the diagnostic objective has been identified, goal tree for the 2X4 LOOP which is shown in Figure 9 was developed; then indicators used for testing the goals were selected. The goal tree is complete in terms of representing the plant objective under consideration, but not all the goals are equally important and useful in on-line diagnosis application. Certain goals in the goal tree such as "Provide Temperature Trip To Prevent Over Heating Heater Rods" and "Provide Heat Exchanger" were assumed to be true. For certain goals the indicators required for testing their success would require calculated parameters that had to be defined using mathematical equations which were not sufficiently simple to be practical. One such case is goal 6. There is no simple equation that can be installed as a calculated parameter for representing the coolant heat removal capability for the various temperatures and pressures. In such situations no indicators are given as long as indicators can be identified for the subgoals.

4.3.2 Goal Success Indicators

The indicators required for testing goal success can be either instrument readings which are identified by channel numbers, or calculated parameters which are identified by the parameter names. The indicators for each goal in the goal tree are shown in

Table 1. For goal number 5, the indicator for testing goal success is a calculated parameter, "DELTA-T". DELTA-T defines the temperature difference between the two thermocouple rakes in the reactor core. The success range for the instrument reading and calculated parameters is identified in Table 2, and their locations in the plant are identified in Table 3. In the process of identifying these success ranges it was recognized that instrument readings are not precise. In instrument channels 20 through 24, where power is measured, small negative power readings are possible instrument readings when the heaters are turned off. This operating knowledge was acquired from the operators of the LOOP. These power output readings are used to define goal success for upper-level goals in the GTST. The selected success range for the power output, between 0 and -1, reflects the most stringent success requirement for the upper-level goals. This range indicates that goal success can exist in the upper-level goals only if there is absolutely no possibility of heater over heating, namely if the heaters are off.

Causal and temporal behavior rules were established by studying the systems, reviewing the experimental results, and interviewing expert operators of the facility. Causal behaviors typified by the statement, "If there is a LOCA and the pressurizer level is dropping, then the source of leak is in the primary system" were established and included in the knowledge base.

Similarly, known 2X4 LOOP transient behaviors obtained from previous operating experience were captured as equations and used to project the behavior. An example of such application of known behavior involves the coolant level curves shown in Figure 10. These curves show the liquid level measurement during a LOCA simulation. Based on this Figure, it is recognized that, after the initial pressure-drop, the upper-core coolant level measurement will stabilize. From operating experience, it is known that the heater will not overheat if the calculated upper-core coolant liquid level is greater than 15%. A parameter, "TIME-TO-UNCOVER," was defined using a mathematical equation which calculates the time before upper core coolant liquid level reaches 15%. This parameter, therefore, shows plant temporal behavior and is used to activate series of LOCA rules that advise the operator on the LOCA behavior.

4.4 On-Line Diagnosis Through GOTRES

During on-line operation, GOTRES provides continuous fault diagnosis. The diagnostic scheme uses a communications program which obtains its input from the micro-VAX computer and creates a GOTRES accessible file (Figure 11). Special input/output functions in GOTRES read the file and install the instrument readings in the knowledge-base. During diagnosis, the search is conducted by performing a comparison between actual instrument values and range of values that indicate goal's success.

4.5 Sample Diagnosis

An example showing a diagnostic path which identifies a loss of coolant fault condition is presented in Table 4. Success parameter for each goal are represented by instrument channel numbers and parameter names. Goals 1, 3, and 8 have instrument readings as success indicators. In the case of goal 6, in which a success parameter is not provided, the goal is assumed to have failed, and the subgoals are checked. Goal 13 has the calculated parameter DELTA-P as its success indicator. Upon testing goal 13, which is a lowest level goal of the goal tree, it is recognized that, "Primary coolant is leaking out." The diagnostic process continues with the rules that expand on this conclusion. Rule 8, which is a causal rule, recognized that the pressurizer was not being used in the particular experiment and could not be a source of leakage. Rule 18, which is one of the temporal LOCA rules, advises the operator on the amount of time before the heater rods would be uncovered. The collective advises from goal 13, rule 8, and rule 18 are displayed on the monitor after the complete diagnosis. An example of this display is shown in Figure 12.

Our experience indicates that 7 to 8 seconds elapse from the time when the expert system requests a new set of instrument values for diagnosis to the completion of the diagnosis. Data transfer from the VAX computer to the IBM AT personal computer consumes most of this elapsed time, the actual diagnosis requires only about 1 second. Our current effort concentrates on expanding this system as an integral part of the process control system for performing automatic feedback and for performing corrective actions following a diagnosis. For this purpose the communication time must be reduced by perhaps housing the expert system as part of the data acquisition computer.

5. OTHER APPLICATIONS

The GOTRES code has also been used to determine root-causes of failures. This process is performed off-line. Instead of on-line instrument values, the user enters information prompted by the computer. To find the root-cause of failure for a particular component, it is first necessary to build a goal tree for that component. Typically, the objective for such a tree would be "prevent failure of component . . ." From this the user/analyst develops a hierarchal description of the process or set of goals which must be achieved if failure is to be prevented. The completed goal tree model provides a cause-consequence description of the component and relates the individual failure initiation sites, their failure progression paths and the failure prevention subgoals which are challenged along the way towards failure of the top goal. To infer which branches of the tree were active during a component failure, similar to the previous example, it is necessary for the code to be able to infer which of the goals failed and contributed to the failure of the top goal. This means that each goal must have a set of criteria (success requirements) associated with the goal's success or failure. In this application the success requirements are mostly based on empirical observation of the component prior to and after its failure, and not necessarily measurable parameters obtained from instruments. This information is supplied by the user in the form of yes/no answers to a set of questions posed by GOTRES, each of which relates to the goal success criteria.

When performing root-cause analysis, GOTRES is accessed via a set of menus which provide options for entering, editing, and printing a tree, or using it to perform diagnostics. When a tree is initially built, it is done so interactively, goal by goal, and upon completion represents an expert description of the cause-consequence relationships between failure initiators, failure causes and their propagation paths. As each goal is entered, the analyst is prompted to define a set of questions which, if answered either "yes" or "no", provide unambiguous evidence of the goal state (success or failure). To generate the questions, the analyst must first mentally simulate the failure, then try to imagine the nature of any information which could possibly be generated as a result, and lastly develop a question which elicits a response of "yes" or "no" to confirm or deny the presence of this evidence. Successfully completing these tasks is difficult and requires an in-depth knowledge of ways in which hardware works and how it fails. Examples of these questions developed as part of root-cause analysis for establishing the hypothesis for crack-initiated failure in a feed-water pump are,

- Is there any evidence for high tensile stress conditions?
- Is there evidence for high residual stresses within the material?
- Is there any evidence for chloride intrusion into the environment within which the material operates?

The method of storage of factual data, knowledge-base, and the process of search for problem-solving is conceptually identical to the previous example. That is, GOTRES finds a goal for which there is evidence of failure; the failure is noted, and ultimately GOTRES synthesizes a description of the likely failure propagation paths. From these descriptions the analyst is able to identify the cause of the failure.

Application of GOTRES for root-cause failure detection and analysis has been very successful. We have reported our experience in this regard in reference (16,17). During the construction of various root-cause analysis goal trees, it became apparent that sections are frequently generic in nature and are used repetitively. Sections of the tree used to explore human errors, corrosion, or cracking mechanisms are examples of these generic sections because they appear at the ends of many branches in their role of causal mechanisms for hardware failure. Their repetitive nature makes it desirable that they be described in a single tree (independent expert module) which is accessed repetitively. This means, however, that a data base must be maintained for each application to make it possible to incorporate the results of each application into the overall hardware goal tree evaluation and allow inference of all of the underlying causes of failure. The method of hierarchy of experts suggested by Chandrasekaran (18) is currently being investigated for incorporation into GOTRES. In this method, the concept of "blackboard" is used as a medium through which each independent expert module can "write" its conclusion. When the conclusion of each independent expert is mixed, then a final conclusion regarding potential root-cause of failure can be reached.

6. CONCLUSION

We have reported the development of an expert system shell which is based on the GTST model. An expert system using this shell has demonstrated its capability in handling causal and temporal behavior during on-line operation of a process plant. The system is also able to isolate faults which can be attributed to multiple origins. Because GTST is based on a problem reduction strategy, it is able to diagnose unanticipated faults. This knowledge structure also allows the user to follow the system's reasoning process as well as to question the rationale behind the system's line of reasoning. The modularity of the GTST structure allows addition of rules or knowledge-frames to the expert system to expand and provide more specific identification of failures. In addition to the application described in this paper, the shell has been effectively tested and used in a root-cause fault diagnosis expert system.

REFERENCE

[1] M. R. Genesereth, "The Use of Design Descriptions in Automated Diagnosis," Artificial Intelligence, 24, (1984). 411-436.

[2] M. R. Genesereth, "Diagnosis Using Hierarchical Design Methods," Proc. AAAI, August 1982. 278-283.

[3] W. Hamscher, "Using Structural and Functional Information In Diagnostic Design,"Proc. AAAI, 1983. 152-156.

[4] R. Davis, "Diagnostic Reasoning Based on Structure and Behavior," Artificial Intelligence, 24, 1984. 347-410.

[5] R. Davis, H. Shrobe, W. Hamscher, et al., "Diagnosis Based On Description Of Structure and Function," Proc. AAAI, 1982. 137-142.

[6] T. J. Laffey, W. A. Perkin, and T. A. Nguyen, "Reasoning About Fault Diagnosis with LES," IEEE Expert, Vol.1, Spring 1986. 13-20.

[7] R. Davis, "Reasoning From First Principles In Electronic Troubleshooting," Int.J. Man-Machine Studies, 1983.

[8] R. Milne, "Fault Diagnosis Through Responsibility," Proc. 9th IJCAI, 1985. 423-425.

[9] M. A. Kramer, B. L. Palowitch, Jr., "Expert System and Knowledge-Based Approaches to Process Malfunctions Diagnosis ," AIChE National Meeting (Nov. 1985).

[10] S. H. Rich, V. Venkatasubramanian, "Model-Based Reasoning In Diagnostic Expert Systems For Chemical Process Plants," Computers and Chemical Engineering, Vol.11.2, 1987. 111-122.

[11] R. N. M. Hunt, M. Modarres, " Integrated Economic Risk Management in a Nuclear Power Plant ," Proceedings of the Annual meeting of the Society for Risk Analysis, October 1984, Knoxville, TN.

[12] N. H. Ulerich, G. J. Powers, "On-Line Hazard Aversion and Fault Detection: Multiple Loop Control Example," 1987 Annual AIChE Meeting, November 15-20, 1987, New York, NY.

[13] P. H. Winston, Artificial Intelligence, 2ed., Addison-Wesley Co., Reading, 1984.

[14] P. H. Winston, B. K. P. Horn, LISP, 2nd Ed., Addison-Wesley Co., Reading, 1984. 311-321.

[15] M. DiMarzo, Y. Y. Hsu, et al, "University of Maryland at College Park (UMCP) 2x4 Loop Test Facility," NUREG/CR -4843, vol.1, 1985.

[16] R. N. M. Hunt, et al, "Development of a Root-Cause Analysis Workstation and Its Test Application," to be presented at the 15th INTER-RAM Conference, Portland, OR. June 1988.

[17] I. S. Kim, M. Modarres,"Application of Goal Tree-Success Tree Model As The Knowledge-Base Of Operator Advisory Systems,"Nuclear Engineering and Design J., October, 1986.

[18] B. Chandrasekaran, "Decomposition of Domain Knowledge into Knowledge Sources: The Mox Approach," Proc. 4th National Conference of Canadian Society for Computational Studies of the Intelligence (CSC SI/SCE10), Saskatoon, Canada, May 1982.

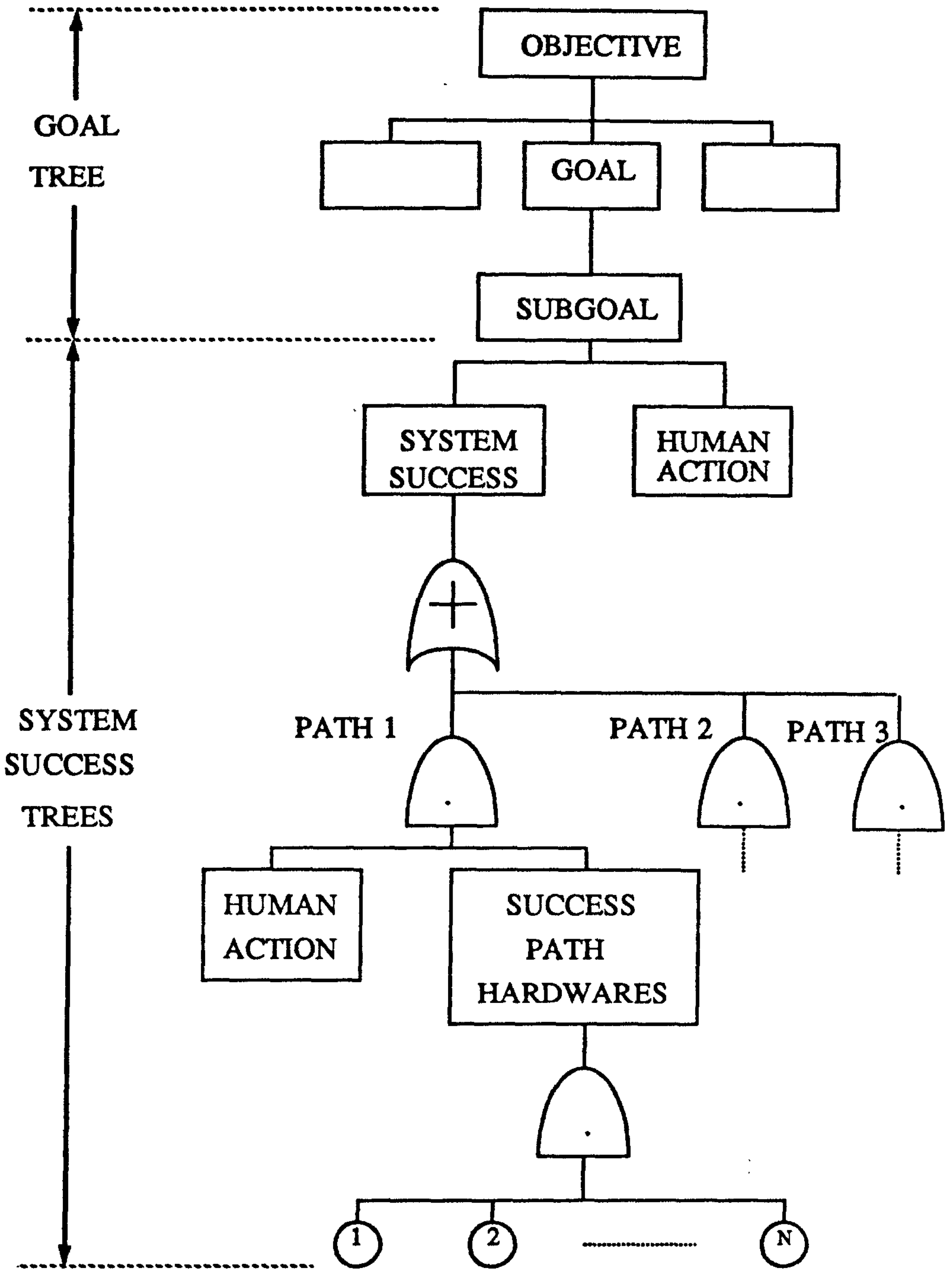


Figure 1. Composite goal tree-success tree structure.

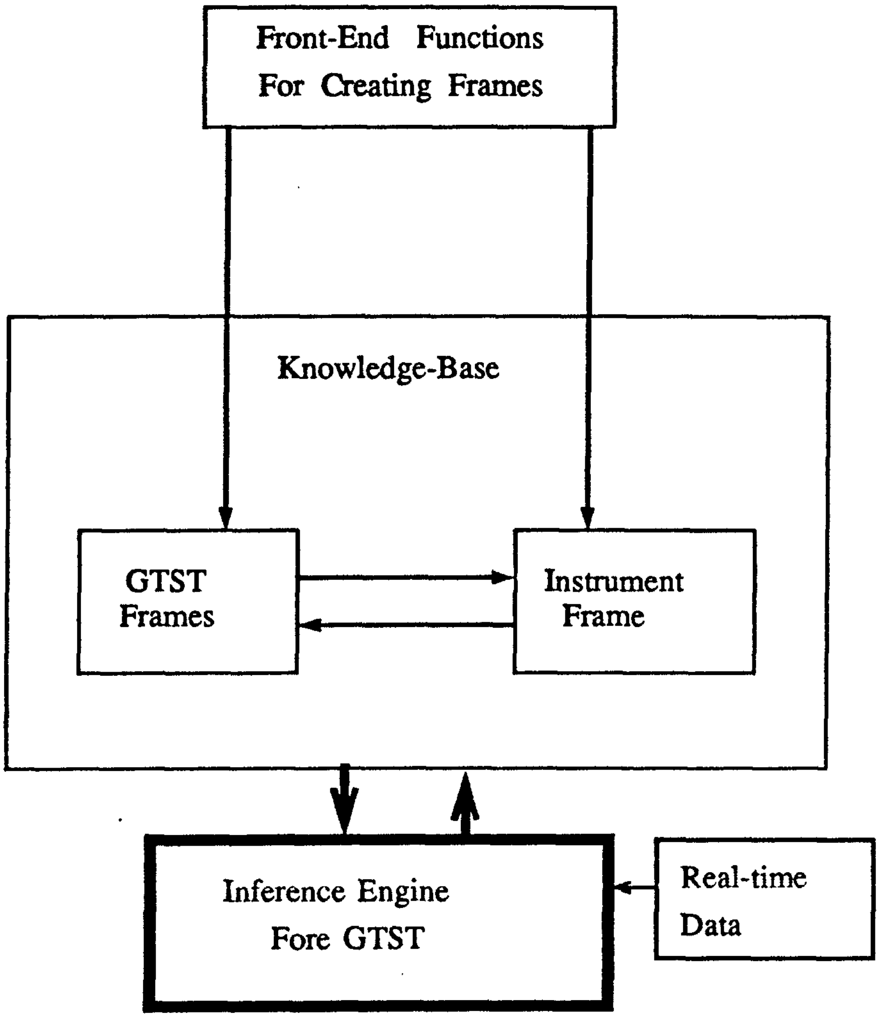
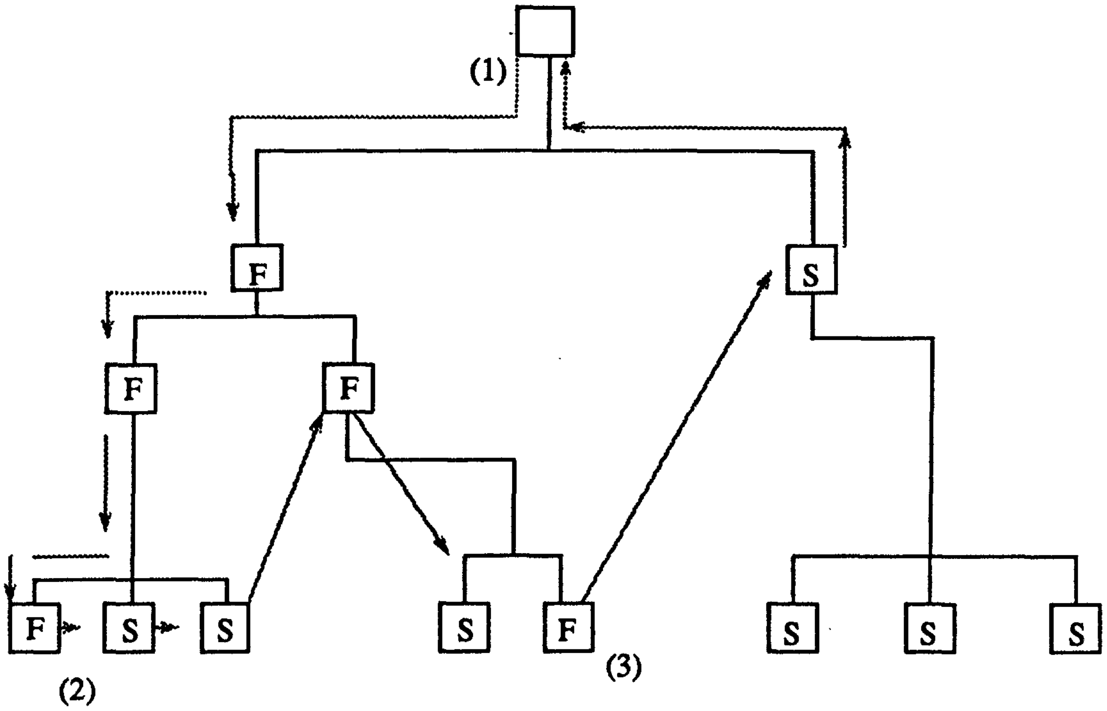


Figure 2 Program structure of GOTRES shell.

GOAL-12

GOAL PATH NAME	*1 2 3 2*
GOAL NAME	*Maintain Sufficient Secondary Coolant*
GOAL PARENT	*Goal-8*
SUBGOALS	*Rule-14* *Rule-15* *Rule-16* *Rule-17*
SUCCESS INDICATORS	*1,6*
VERBAL INDICATORS	
ADVISEMENT	*Secondary Side Coolant Level Is Low*

Figure 3. A frame representation of a goal in GTST.



(1) The search process traces down a failure path.

(2) The process checks other goals that are on the same level for failed goals, then checks on the preceding level and traces down a second cause.

(3) The process checks the remaining goals on the upper level and finds no more failed goals, and the process terminates when the top level is reached.

Figure 4. The flow of the search process used in GOTRES.

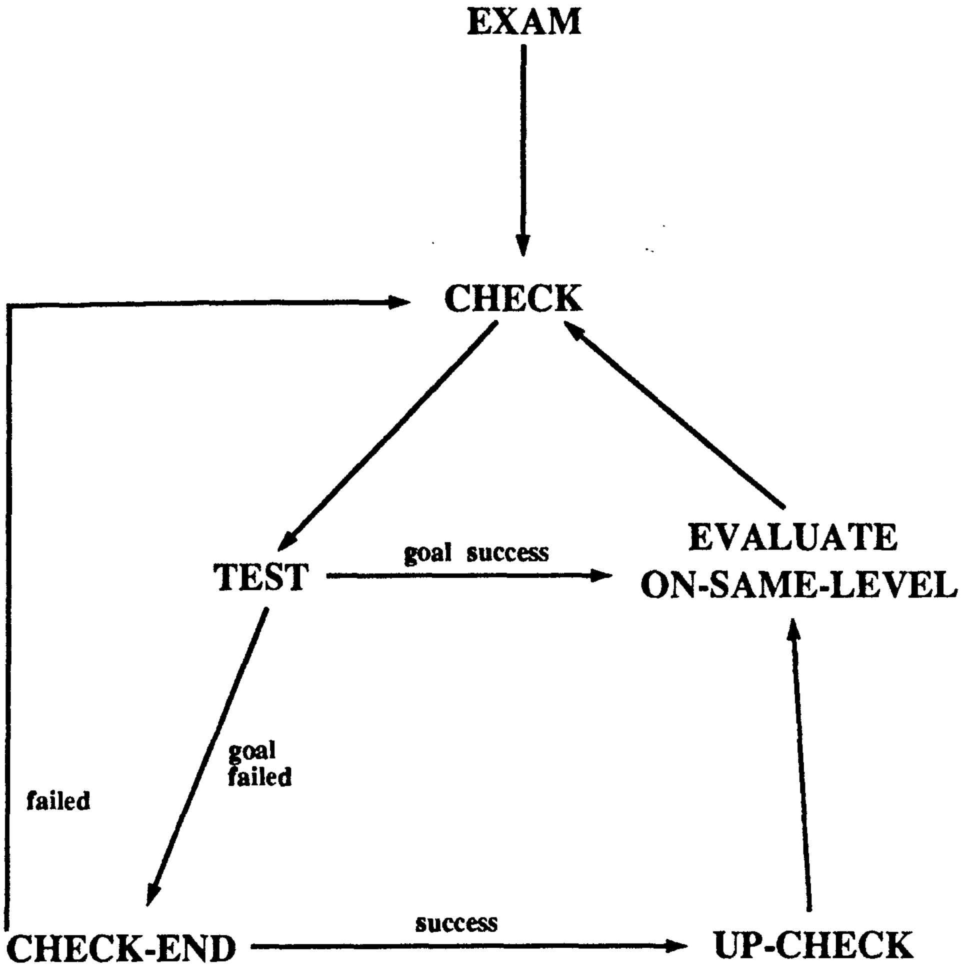


Figure 5. The control structure of GOTRES expert shell.

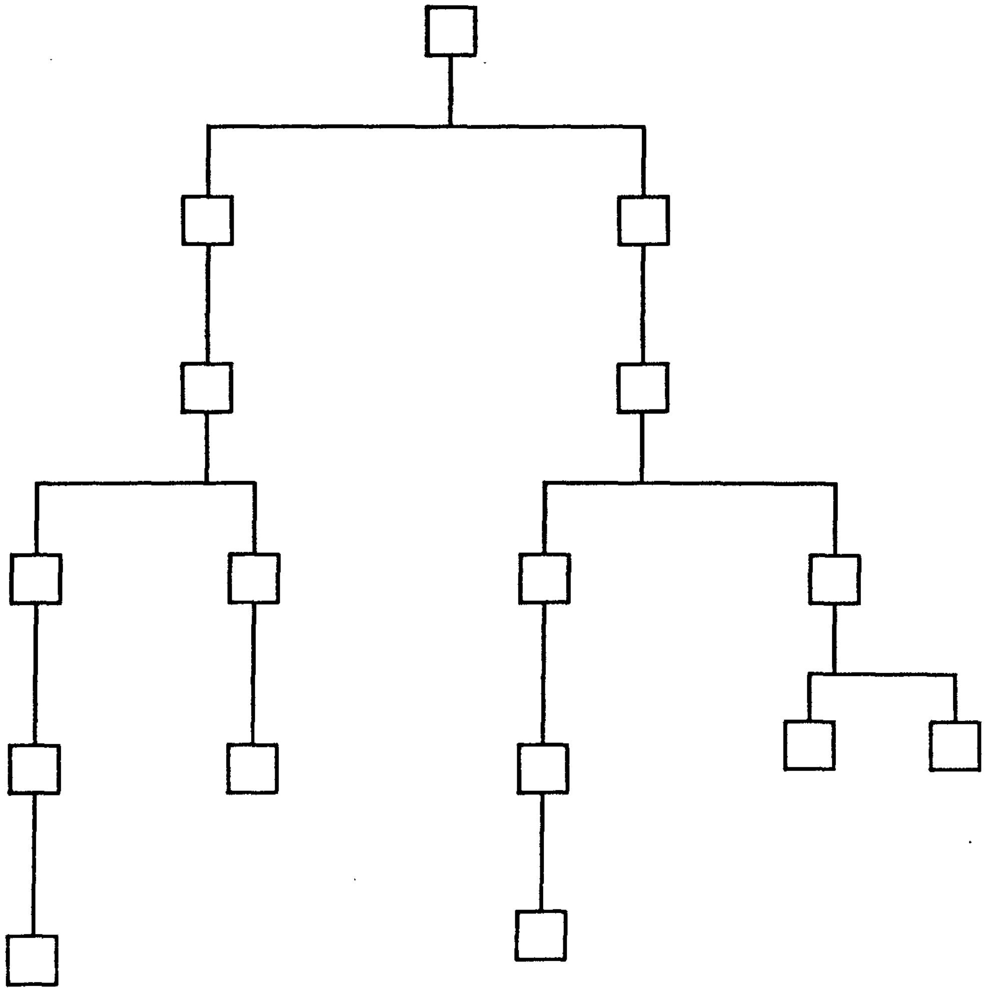


Figure 6. An example of a GTST structure in which breath-search method is more effective.

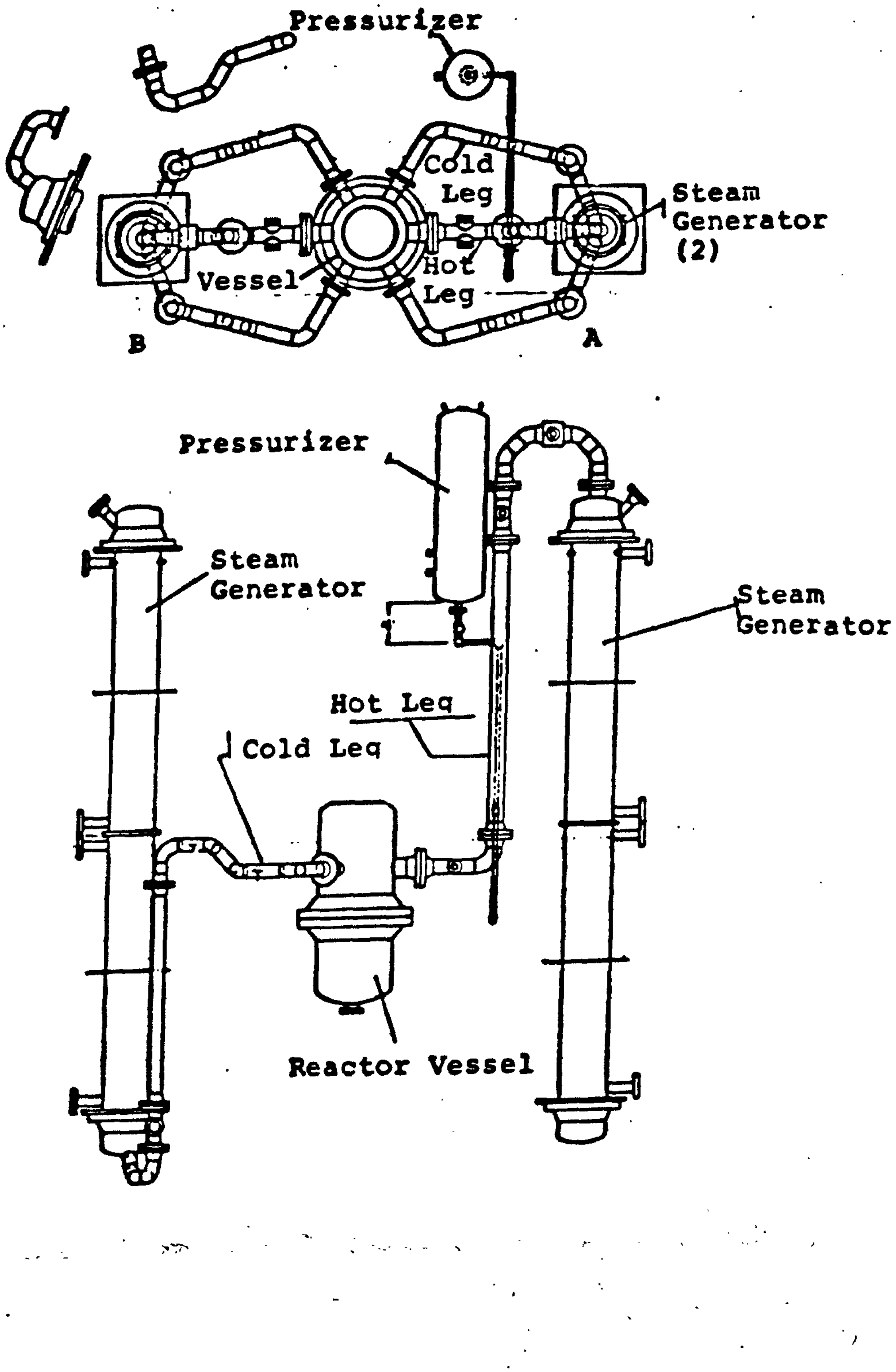


Figure 7. Overall LOOP assembly drewing.

CHANNEL LOCATION OF UMCP 2X4 LOOP

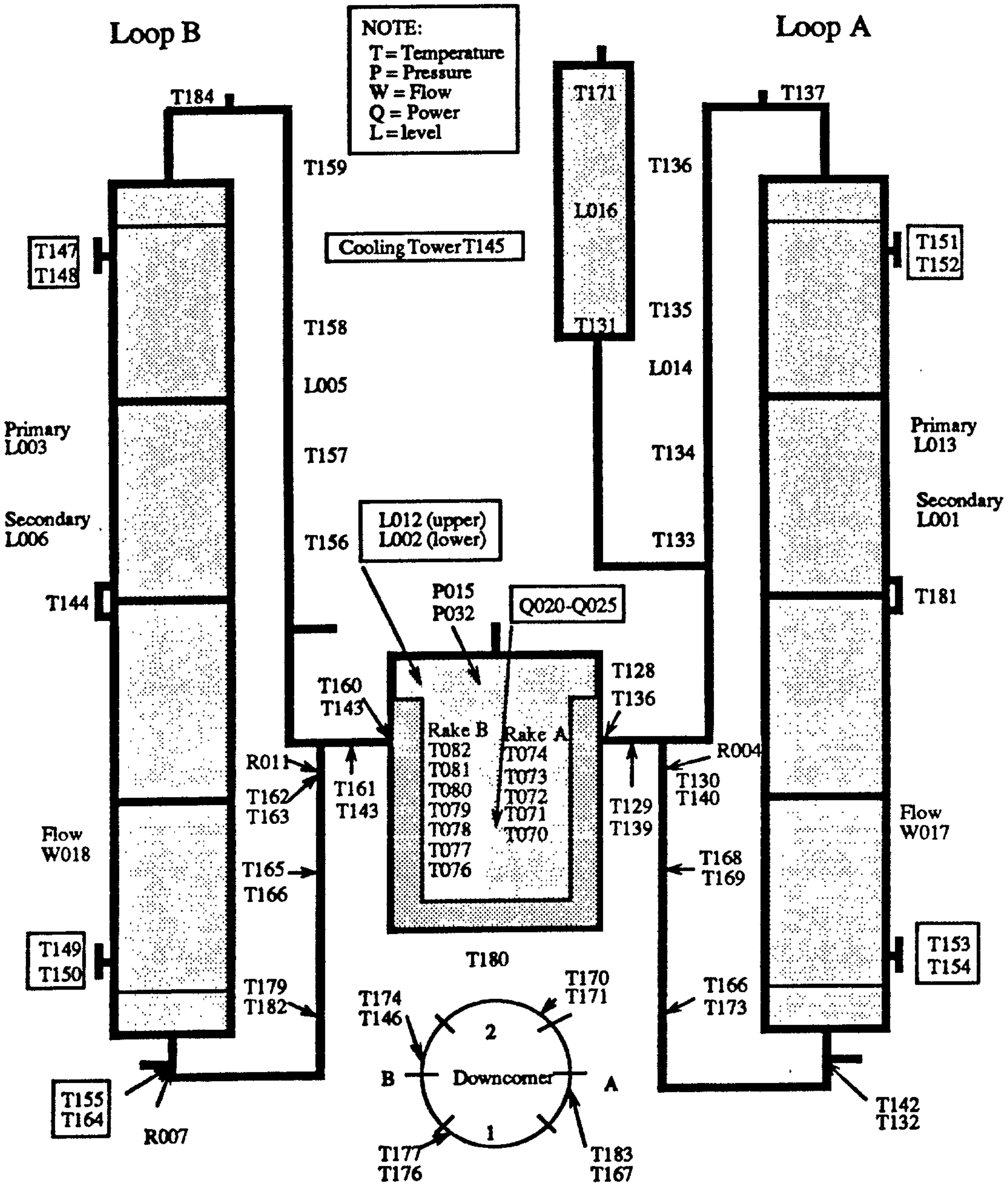


Figure 8. Overall LOOP assembly instrument site.

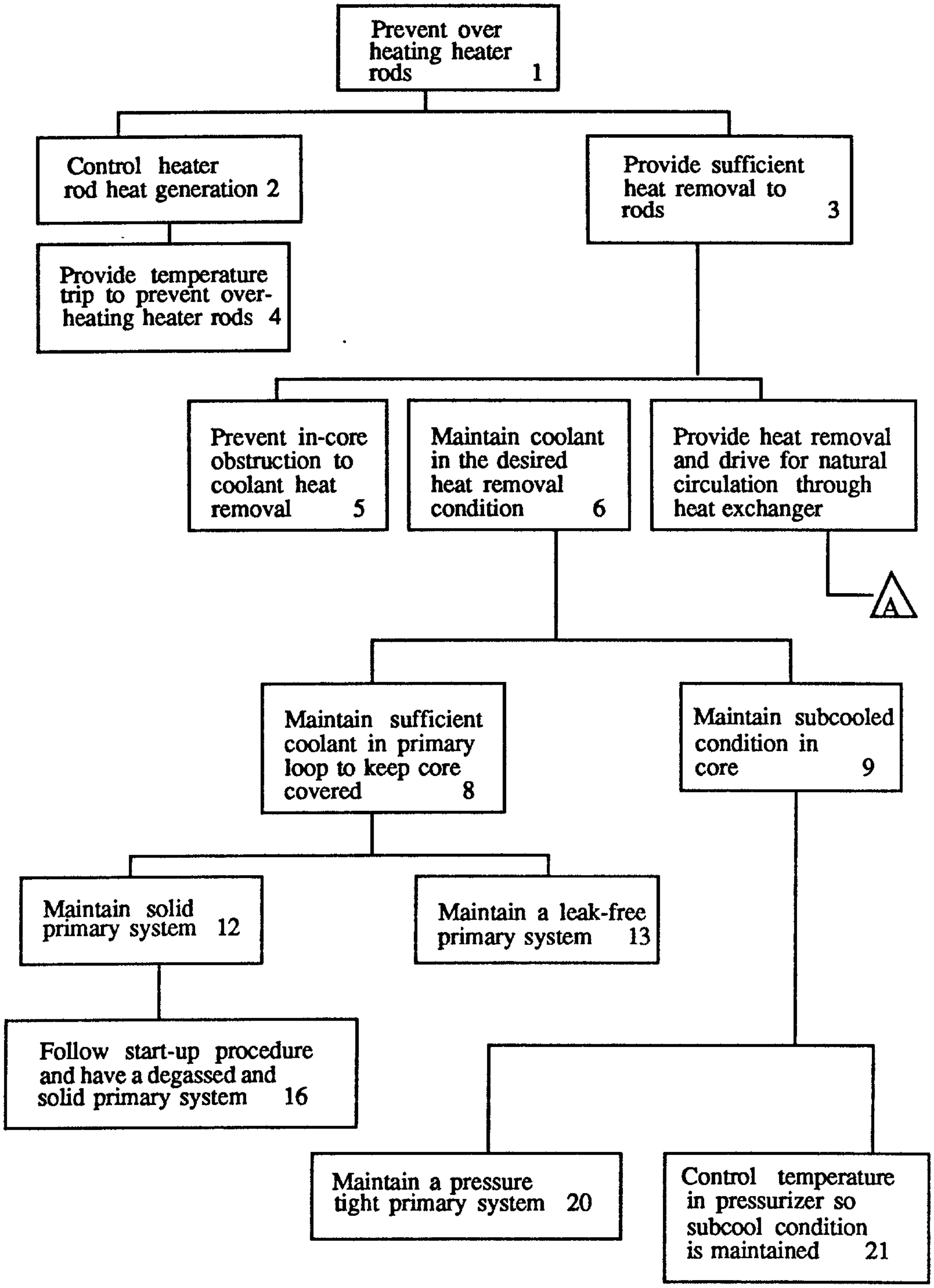


Figure 9. Goal tree for the University of Maryland Loop.

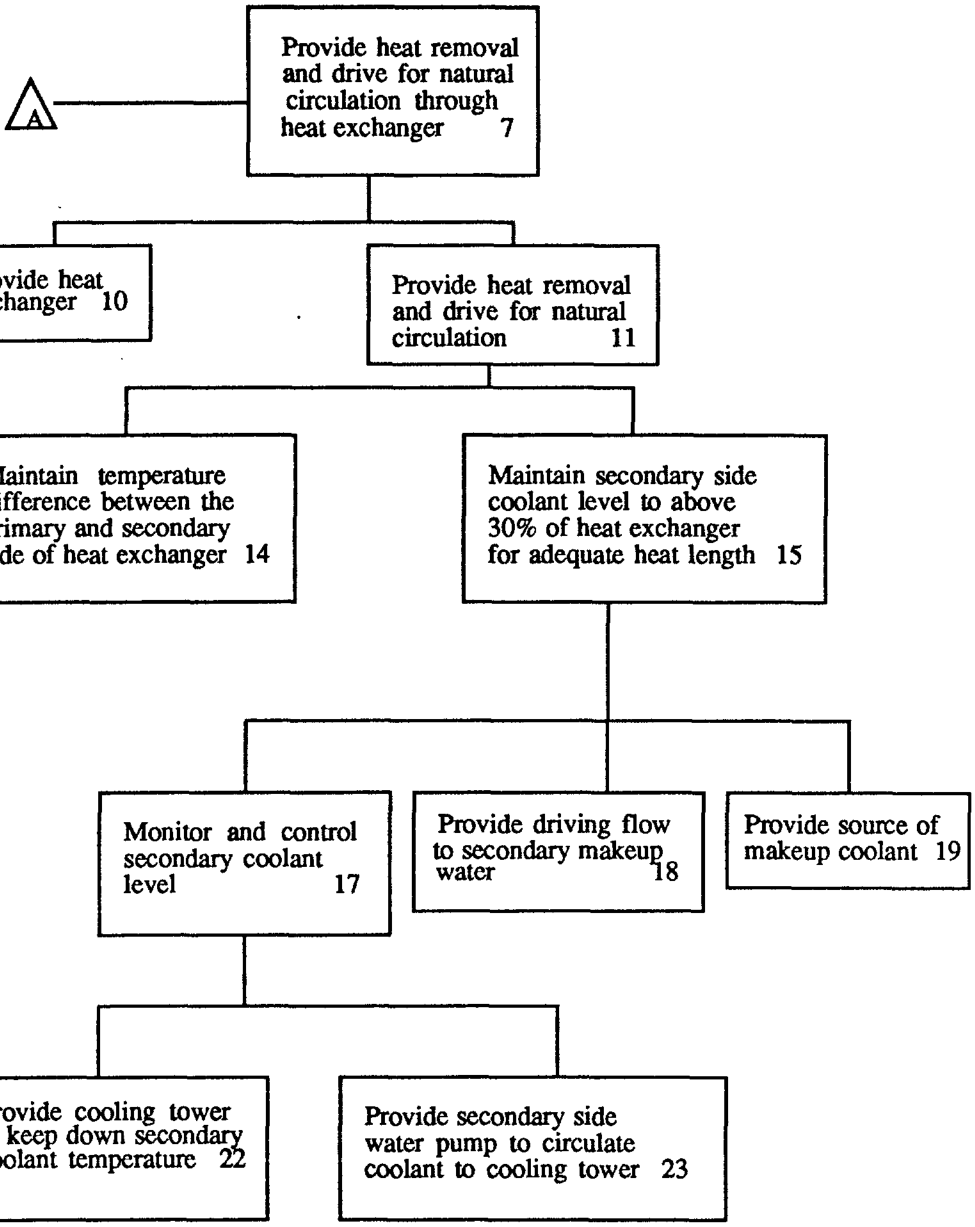


Figure 9. Continued.

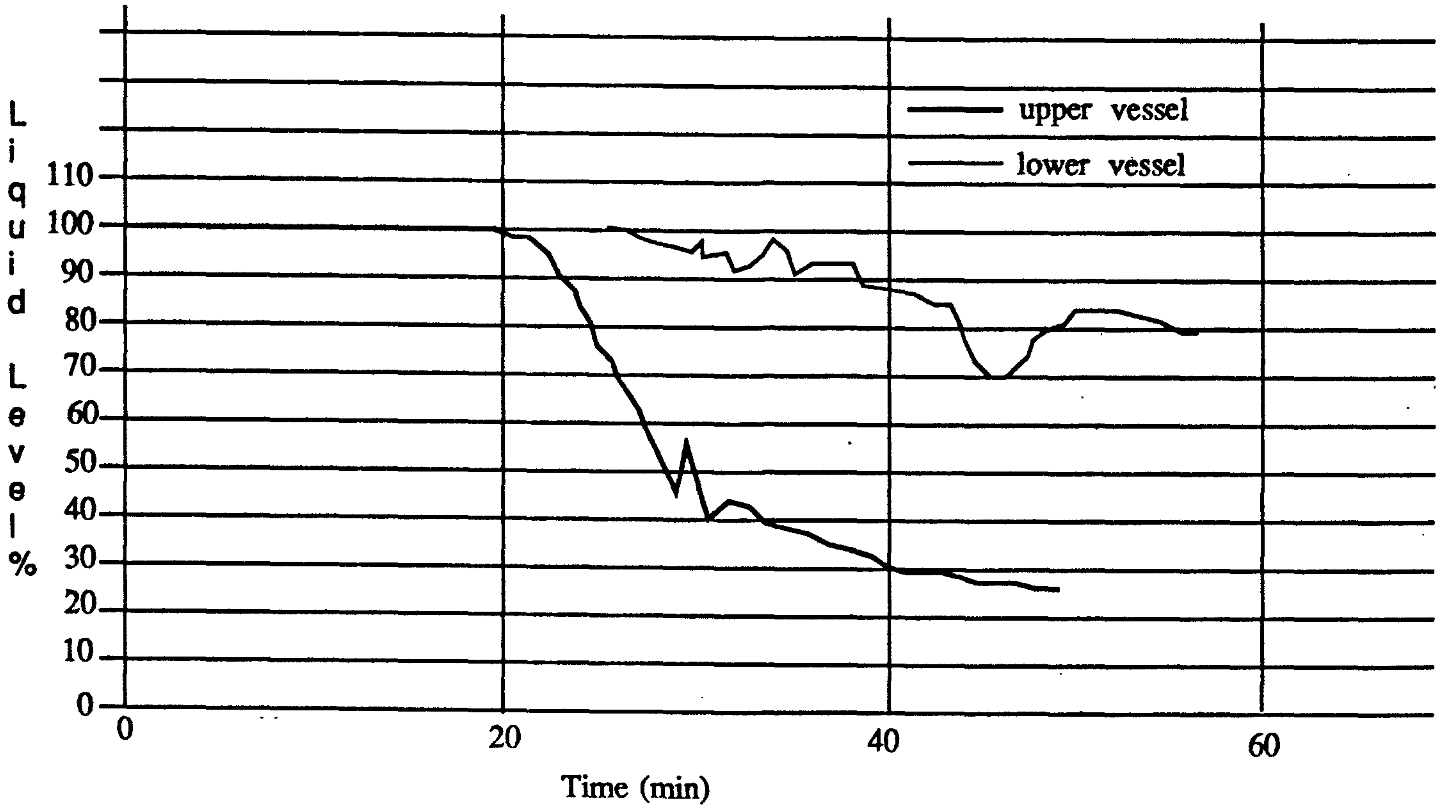


Figure 10. Plant loss-of-coolant behavior as a function of time.

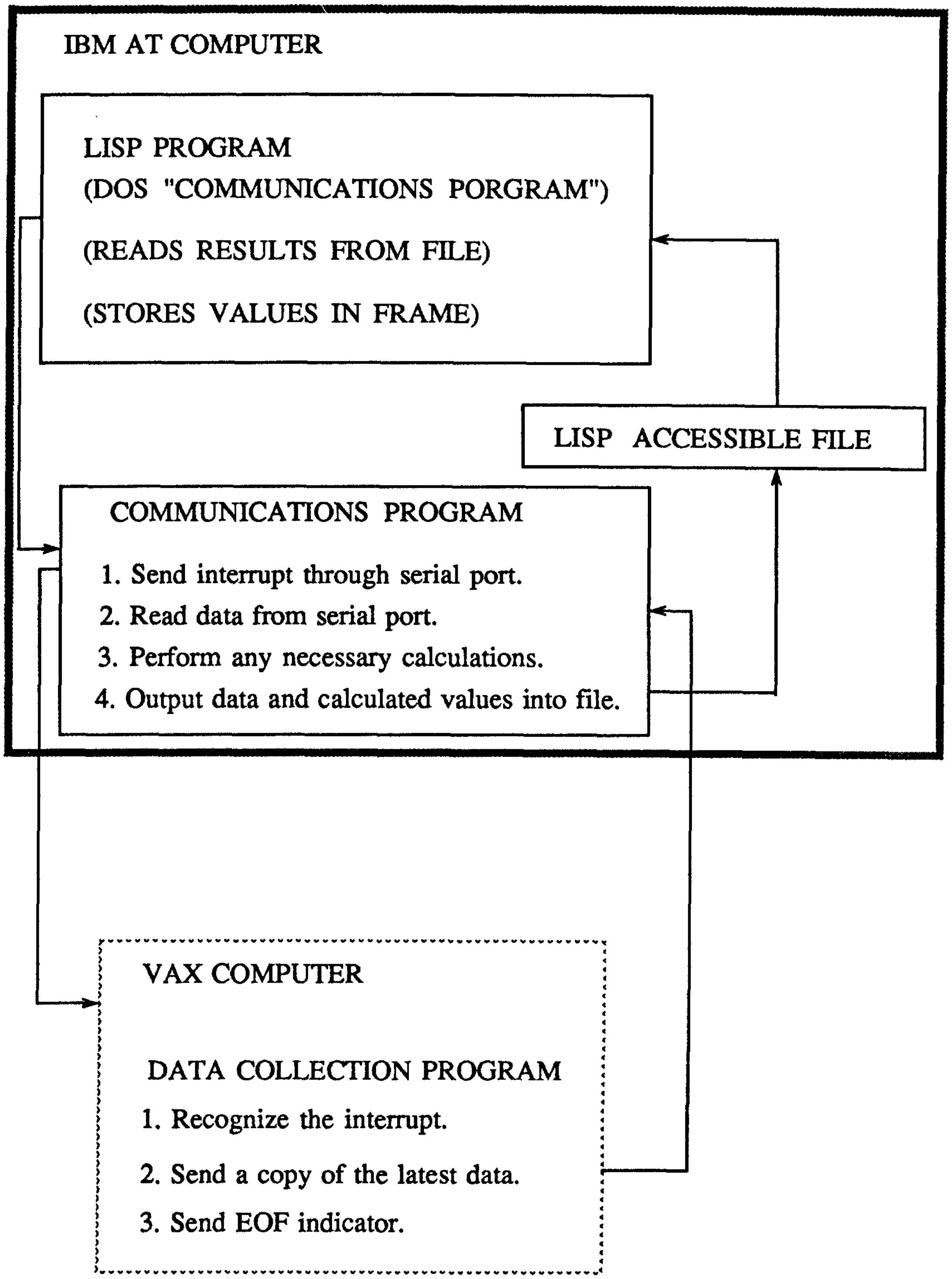
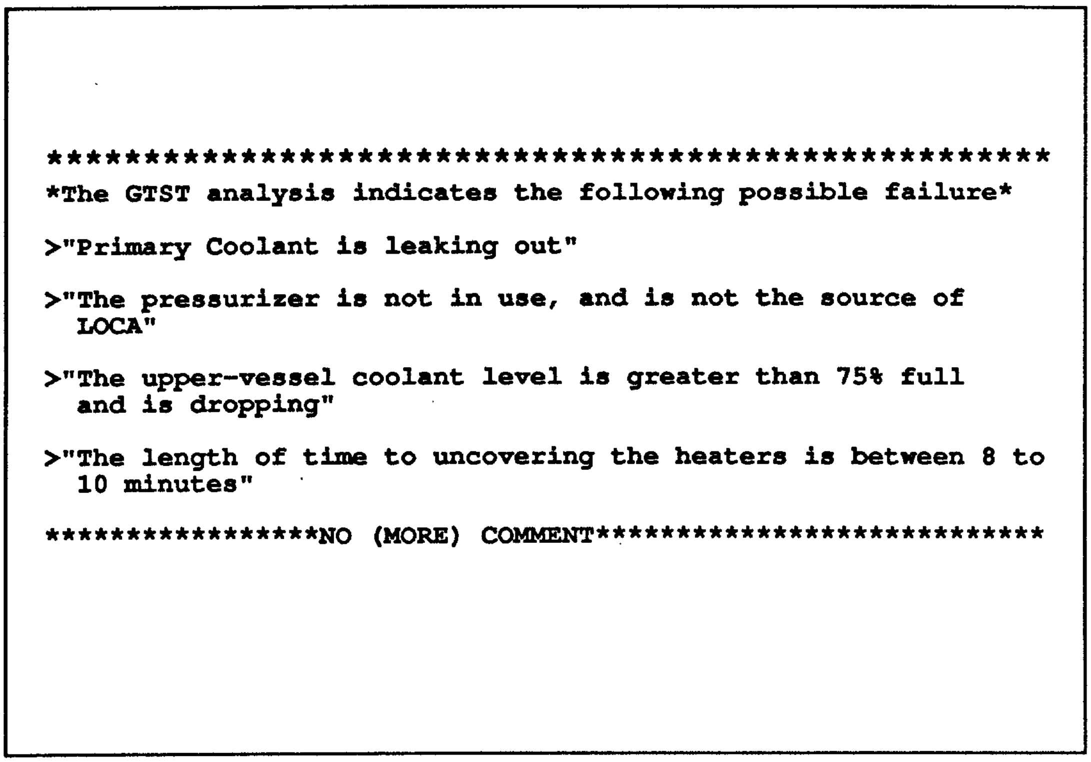


Figure 11. Schematic of data flow in GOTRES during on-line fault diagnosis.



```
*****************************************************************
*The GTST analysis indicates the following possible failure*

>"Primary Coolant is leaking out"

>"The pressurizer is not in use, and is not the source of
  LOCA"

>"The upper-vessel coolant level is greater than 75% full
  and is dropping"

>"The length of time to uncovering the heaters is between 8 to
  10 minutes"

*****************NO (MORE) COMMENT*****************************
```

Figure 12. The GOTRES display of diagnostic results.

GOAL NUMBER	GOAL STATEMENT	INSTRUMENT CHANNEL # AND PARAMETER NAMES
1	Prevent over heating heater rods	70
2	Control heater rods heat generation	20, 21, 22, 23, 24, 25
3	Provide sufficient heat removal to rods	74, 81
4	Provide temperature trip to prevent over-heating heater rods	---
5	Prevent in-core obstruction to coolant heat removal	DELTA-T
6	Maintain coolant in the desired heat removal condition	---
7	Provide heat removal and drive for natural circulation through heat exchangers	13
8	Maintain sufficient coolant in primary loop to keep core covered	12
9	Maintain subcool condition in core	81
10	Provide heat exchanger	---

Table 1 Goal number and success indicators.

GOAL NUMBER	GOAL STATEMENT	INSTRUMENT CHANNEL # AND PARAMETER NAMES
11	Provide heat removal and drive for natural circulation	3, 13
12	Maintain solid primary system	71, 72
13	Maintain a leak-free primary system	DELTA-P
14	Maintain temperature difference between the primary and secondary side of heat exchanger	THERMAL-1
15	Maintain secondary side coolant level to above 30% of heat exchanger for adequate heat length	1, 6
16	Follow start-up procedure and have a degassed and solid primary system	---
17	Monitor and control secondary coolant level	---
18	Provide driving flow to secondary makeup water	26, 27
19	Provide source of makeup coolant	17, 18

Table 1 CONTINUE.

GOAL NUMBER	GOAL STATEMENT	INSTRUMENT CHANNEL # AND PARAMETER NAMES
20	Maintain a pressure tight primary system	15
21	Control temperature in pressurizer so subcool condition is maintained	THERMAL-2
22	Provide cooling tower to keep down secondary coolant temperature	16, 145
23	Provide secondary side water pump to circulate coolant to cooling tower	---

Table 1 CONTINUE.

INSTRUMENT CHANNEL #	INSTRUMENT	UNITS	UPPER BOUND	LOWER BOUND
1	DP Cell	%Level	100	30
2	DP Cell	%Level	100	90
3	DP Cell	%Level	100	40
5	DP Cell	%Level	100	5
6	DP Cell	%Level	100	30
12	DP Cell	%Level	100	40
13	DP Cell	%Level	100	30
16	DP Cell	%Level	100	30
17	Flow Meter	kg/s	2	0.01
18	Flow Meter	kg/s	2	0.01
20	RMS Meter	KW	0	-1
21	RMS Meter	KW	0	-1
22	RMS Meter	KW	0	-1
23	RMS Meter	KW	0	-1
24	RMS Meter	KW	0	-1
25	RMS Meter	KW	0	-1
26	Boil-off Box, A (on/off)	Volt	2	0.1
27	Boil-off Box, B (on/off)	Volt	2	0.1
32	Pres. Tran.	Kpa	7	1
70	Thermocouple	C	100	0
71	Thermocouple	C	100	0
72	Thermocouple	C	100	0
74	Thermocouple	C	100	0
76	Thermocouple	C	100	0
77	Thermocouple	C	100	0
81	Thermocouple	C	100	0
145	Thermocouple	C	140	20

Table 2 Instruments and their success ranges.

INSTRUMENT CHANNEL #	INSTRUMENT	LOCATION
1	DP Cell	Steam Gen. A, Secondary Side
2	DP Cell	Lower Vessel
3	DP Cell	Steam Gen. B, Primary Side
5	DP Cell	Hot Leg B
6	DP Cell	Steam Gen. B, Secondary Side
12	DP Cell	Upper Vessel
13	DP Cell	Steam Gen. A, Primary Side
16	DP Cell	Pressurizer
17	Flow Meter	Secondary Side, A
18	Flow Meter	Secondary Side, B
20	RMS Meter	Vessel Heater (power cab. #1)
21	RMS Meter	Vessel Heater (power cab. #2)
22	RMS Meter	Vessel Heater (power cab. #3)
23	RMS Meter	Vessel Heater (power cab. #4)
24	RMS Meter	Vessel Heater (power cab. #5)
25	RMS Meter	Vessel Heater (power cab. #6)
26	Boil-off Box (on/off)	Boil-off Pump, A
27	Boil-off Box (on/off)	Boil-off Pump, B
32	Pres. Tran.	Vessel (ll)
70	Thermocouple	Vessel Rake #1, Location 1
71	Thermocouple	Vessel Rake #1, Location 2
72	Thermocouple	Vessel Rake #1, Location 3
74	Thermocouple	Vessel Rake #1, Location 5
76	Thermocouple	Vessel Rake #2, Location 1
77	Thermocouple	Vessel Rake #2, Location 2
81	Thermocouple	Vessel Rake #2, Location 5
145	Thermocouple	Coolant Tower

Table 3. Instruments and their location.

GOAL #	GOAL STATEMENT	SUCCESS INDICATOR
1	Prevent over heating heater rods	70
3	Provide sufficient heat removal to rods	74, 81
6	Maintain coolant in the desired heat removal condition	---
8	Maintain sufficient coolant in primary loop to keep core covered	12
13	Maintain a leak-free primary system	DELTA-P

Rule	ADVISEMENT TO OPERATOR	SUCCESS INDICATOR
8	"The pressurizer is not in use, and is not the source of LOCA"	TIME-TO-UNCOVER
18	"Time to uncovering the heater rods is greater than 10 minutes"	TIME-TO-UNCOVER

Table 4 An example search path and the advicement provided to the operator.