Submitted to and accepted for the Expert Systems in Production and Operations Management Conference, Hilton Head, South Carolina, 1990

TEST+: The Extension of an Application Shell For Turbine-Generator Diagnosis

Bruce M. McLaren and Gary S. Kahn
Carnegie Group, Inc.
Five PPG Place
Pittsburgh, Pa. 15222

Abstract

As a variety of techniques and problem-solving methods become available for expert systems, practitioners need to understand how specific domain and task characteristics should influence the selection of methods. The diagnostic task domain varies with respect to many attributes. These include the amount and quality of data, degree of access to component failures, conclusiveness of test results, and decision-making complexity.

Recognition of these differences leads to the specific architectural features of TEST+. TEST+ provides a variant of the TEST architecture to address reasoning under uncertainty and decision complexity. Integrating the strengths of inexact, evidential systems with the model-based approach of TEST, TEST+ provides a unique and powerful solution to the diagnosis of turbine generators.

1 Introduction

As a variety of techniques and problem-solving methods become available for realizing expert systems, a stronger engineering foundation is needed to further the art. In particular, practitioners of applied systems need grounding in how to recognize when particular approaches are motivated by specific domain and task characteristics. To some extent this has been recognized in attempts to associate problem-solving methods, such as heuristic classification, with tasks, such as diagnosis (Clancey [1]; McDermott [6]). However, successful engineering requires a more fine-grained analysis of domain and task characteristics. This is nowhere more true than for diagnostic systems. Since the MYCIN experiments (Shortliffe [8]), many different heuristic, model-based, and hybrid approaches have been reported. For review articles see (Hayes-Roth [3]) and relevant chapters of (Shrobe [9]).

In order to understand how domain and task characteristics motivate particular system functions and architectures, this paper discusses TEST+. Although this analysis is no substitute for a comprehensive analysis of diagnostic techniques, it does provide a step forward and demonstrates the kind of evaluation required. (Kahn [4]) provides another example.

TEST+ is a variant enhancement of the TEST architecture (Kahn et al. [5]). In what follows we describe the origins of TEST+ in the turbine-generator domain. Features of this domain motivated a particular set of system requirements that were met by modifying the TEST architecture. The result, TEST+, is described and evaluated.

2 The Origins of TEST+

Although TEST can be characterized as a domain-independent troubleshooting shell, the attempt to apply TEST outside its domain of origin (vehicle diagnosis) has had mixed success. In some cases, TEST works very well (Seligman et al. [7]); and in others, limitations become apparent. The attempt to apply TEST to the diagnosis of turbine generators led to TEST+.

The TEST+ architecture is motivated by critical features which distinguish the turbine-generator domain from that of vehicle diagnosis and others to which TEST applies:

- Diagnosis is based on voluminous and uncertain information;
- There is limited and difficult access to components which fail;
- There is a lack of conclusiveness in the significance of evidence as it bears on any one hypothesis; and
- There is considerable complexity in key diagnostic decisions.

Vehicle diagnosis is typically based on key symptomatic occurrences, together with the results of confirmatory checkout and test procedures. In contrast, turbine generators must be diagnosed on the basis of difficult to interpret, copious, and, in some cases, obscure vibration data, manufacturing specifications, plant state data, and historical information.

Unlike vehicle engines that can be easily disassembled or directly tested, turbine generators must, for the most part, be maintained in operation during diagnostic evaluation. Turbine generators weigh several hundred tons and are responsible for generating electricity for millions of utility users. Given the cost of lost generator operation, plant experts avoid bringing down machines for tests and repairs.

Data available to turbine-generator technicians is typically far less conclusive than that available to their automotive counterparts. Because of the potential occurrence of multiple faults, the lack of immediate test data, as well as the factors mentioned above, experts in this domain are more cautious in drawing conclusions prior to evaluating a large number of potential candidate explanations.

Finally, there is considerable complexity in deciding which faults should be investigated further and which tests run. Turbine-generator experts typically make qualitative decisions, involving substantial financial and safety considerations, regarding the continued operation of machinery which might require repair or preventive maintenance. Management desires maximum utilization to maximize profit; the experts must decide when to trade-off utilization for requisite repair and maintenance.

TEST does not adequately address these domain characteristics. For one, TEST provides no way to represent either uncertain data or inconclusive implications. Because TEST evolved from domains in which a small amount of certain data could support diagnostic conclusions, it also provides no mechanism for explicitly combining multiple sources of uncertain information. Since automotive technicians had relatively good access to component failures, TEST quickly targets conclusive tests and avoids evaluation of alternative explanations. Finally, because of relatively simple considerations behind diagnostic decision-making in the garage, TEST provides little flexibility to the end user in terms of controlling the diagnostic process. TEST+ provides enhancements in each of these areas.

3 The TEST+ Architecture

3.1 The TEST Foundation

TEST+ utilizes and builds upon the representational structure and problem-solving method of TEST. The TEST architecture is based on a distinction between a domain-independent problem solver (inference engine) and domain-specific knowledge bases. A TEST knowledge base is composed of instances of prototype objects connected via predefined relationships. The failure mode is the central object in the knowledge base. A failure mode represents a state of deviation from standard operation for the Unit Under Test (UUT). The failure modes in a TEST knowledge base are hierarchically organized and attached to one another via due-to, or causal links. The failure modes which reside at the top of the hierarchy represent observable or symptomatic failure states of the UUT. Successive intermediate failure modes at lower levels provide a "backward chain" through due-to links to root causes. Other objects, such as questions, tests, and repair procedures, are associated with failure modes.

TEST knowledge bases may include rules which have meta-level effects – i.e., the alteration of the knowledge base during diagnosis. Rules may alter the order of investigation of failure modes, add or delete failure modes from the knowledge base, or alter attribute (or procedural) information associated with other objects (e.g., repairs, tests) found in the knowledge base.

To perform diagnosis, in the default case, TEST executes a recursive depth-first search on the hierarchical, causal network, commencing with the symptomatic or observable failure modes which reside at the top of the hierarchy. Failure modes exist in one of three states: confirmed, disconfirmed, or unknown. TEST explores the possible causes of a confirmed or unknown failure mode, call it FOCUS-FM, by sequentially evaluating the failure modes listed in its due-to relational attribute. As soon as a failure mode in this list becomes either confirmed or unknown, TEST recursively repeats the process by using the newly confirmed/unknown failure mode as the new FOCUS-FM. Evaluation of a failure mode (i.e., assignment of confirmation/disconfirmation/unknown) is achieved through direct tests and/or evidentially conclusive rules. The search concludes when a FOCUS-FM is confirmed which has no causally related failure modes, i.e., it is a root cause.

3.2 TEST+ Enhancements

TEST+ builds on TEST in three ways. It provides for the representation of uncertainty; modifies the search algorithm; and gives users more visibility into and control over the diagnostic process.

Within TEST+, failure modes may be assigned confidence factors (CF) on the basis of evidential or "uncertain" rules. Following the MYCIN approach (Shortliffe [8]), these rules contribute a measure of belief (MB) or disbelief (MD) to a hypothesis (failure mode) with which they are associated. TEST+ allows rules to have Boolean antecedents which may refer to known data or evoke questions and tests.

Unlike MYCIN, TEST+ links rules into the semantic net of failure modes and other objects provided by TEST. A typical rule in the turbine-generator knowledge base is

> *If the dominant frequency is 1f and the constant of change is greater than 10 minutes but less than or equal to half an hour, then there is moderate evidence (.7) for the hydrogen seal failure mode.*

This rule would be represented as an object linked to the hydrogen seal failure mode. Each failure mode can have multiple rules of this kind associated with it. The contributions of multiple rules bearing on the same failure are aggregated, as in MYCIN.

Evidential rules are evaluated only when TEST+ focuses on a failure mode with which they are associated. As in TEST, search commences by focusing on an observable failure (or symptom) which dominates a hierarchy of intermediate and root causes. However, while TEST pursues its evaluation depth-first, TEST+ uses a novel approach – localized best-first with backtracking.

This search algorithm proceeds as follows, given a failure mode, FOCUS-FM (see above).

1. If FOCUS-FM is a root cause (terminal failure mode), go to 5.
2. For each failure mode linked to FOCUS-FM through the due-to relation, assign a CF, by evaluating it's associated evidential rules.
3. Of these, set FOCUS-FM to the failure mode with the highest CF greater than or equal to the global certainty threshold (GCT). This is a preset (but alterable) global variable.
4. If there is such a failure mode, go to 1, otherwise …
5. If there are no unevaluated failure modes, go to 8.
6. Display all failure modes with a CF >= GCT and ask the user to choose the next nonterminal failure mode to pursue or to have TEST+ select a candidate. TEST+ will select the nonterminal failure mode, of those it has already evaluated, with the highest CF >= GCT.
7. Set FOCUS-FM to this selection. Go to 2.
8. Display all failure modes with their associated CF.

An example clarifies the strategy.

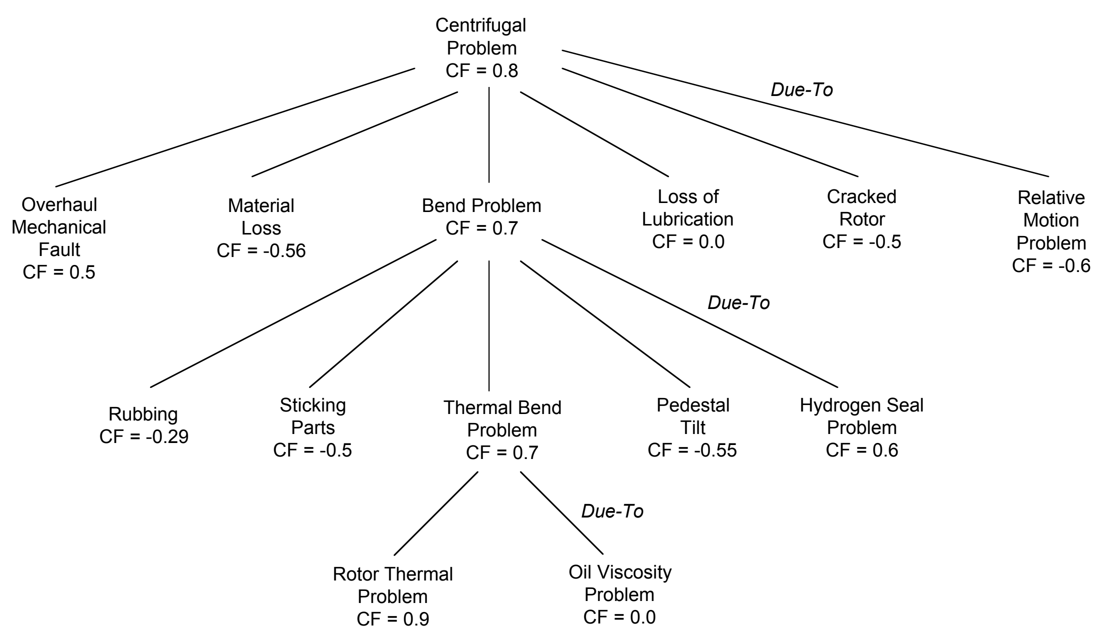

Figure 1: A Portion of the Turbine-Generator Causal Network

Assuming a GCT of 0.5 and an initial FOCUS-FM of "Centrifugal Problem" (see Figure 1), the search would proceed as follows. Evaluation of the possible causes of the "Centrifugal Problem" assigns "Bend Problem" to be the new FOCUS-FM, as it has the CF with the highest value greater than or equal to the GCT. Notice that the failure

"Overhaul Mechanical Fault" remains in consideration for future investigation as its CF is equal to the GCT. Next the problem solver evaluates the underlying causes of the "Bend Problem" and chooses "Thermal Bend Problem" as the new FOCUS-FM. Once again a failure which was not chosen from the list of causes, the "Hydrogen Seal Problem," remains in consideration due to its CF being greater than the GCT. Evaluation of the underlying causes of "Thermal Bend Problem" arrives at a high degree of belief in the root cause "Rotor Thermal Problem." At this juncture the system has reached a decision point. The user may accept the strong belief in the "Rotor Thermal Problem" as a final diagnosis and discontinue the session or may continue the search by backtracking. Since both "Overhaul Mechanical Fault" and "Hydrogen Seal Problem" have CFs >= GCT, the user may choose one of these intermediate causes for continued search or may allow the problem solver to select its favored choice, "Hydrogen Seal Problem," for continued search.

At any point during diagnosis the user is able to view the current global state of the diagnostic session. The global state consists essentially of a categorization of all of the failure modes encountered during the session. Failure modes are categorized after evaluation as (a) possible root causes (i.e., terminal nodes with CFs >= GCT) (b) discounted root causes (i.e., terminal nodes with CFs < GCT) (c) possible intermediate causes (i.e., nonterminal nodes with CFs >= GCT) or (d) discounted intermediate causes (i.e., nonterminal nodes with CFs < GCT). The distinction between root and intermediate causes is provided because root causes elicit heightened attention of the diagnosticians. At any point during a diagnostic session, the user may alter the GCT, causing a possible recategorization of the uncertain failure modes and a refocusing of the problem solver. As final output of a session, the user is presented with a list, ordered by descending CF values, of all the possible root causes of the initiating symptom.

4 Design Decisions Considered

The TEST+ architecture is the result of a number of design decisions. This section considers those bearing on the representation of uncertainty, breadth of evaluation or search, and user control.

4.1 Representing Uncertainty

TEST+ takes advantage of rule-based techniques for handling uncertainty, but uses TEST's model-based approach to avoid the practical difficulties that arise with pure EMYCIN-type systems. The EMYCIN approach to representing and composing certainty factors has proved to be of wide applicability. In fact, the PDS system (Fox et al. [2]) had already taken this approach to the diagnosis of turbine generators with some success.

Practical difficulties, however, typically arise with EMYCIN applications. In particular, exhaustive search through large problem spaces is ineffective for interactive use; knowledge acquisition is hindered as experts tend to reason with both heuristic rules and causal knowledge; and large systems are difficult to maintain since there is a lack of modularity when implications propagate through large rule sets.

TEST+ provides an effective remedy to these problems. TEST+ avoids exhaustive search by focusing on the candidates best supported by the evidence as it accumulates. Relying on the causal structure of TEST knowledge bases as a backbone, TEST+ gains an enriched set of concepts for knowledge acquisition (causal relations and heuristic rules), as well as increased modularity. Avoiding large rule sets, TEST+ clusters rules around each failure mode. These rules are eligible to fire only when the failure mode with which they are associated is being evaluated by the problem solver. Maintainability is enhanced by further restricting the scope of rules. CFs which result from antecedent satisfaction are only propagated locally; i.e., a failure mode's CF will have no affect on another failure mode's CF.

4.2 Search

Given the inaccessibility of fault states during operation, and the relative inconclusiveness of available data, TEST+ evaluates a set of plausible candidates. However, because of the large number of potential faults it avoids exhaustive evaluation. This is achieved with a best-first search where the goodness of a node is reevaluated as more evidence is acquired. Unless overridden by the user, TEST+ will consider all avenues of explanation that are above the global threshold before terminating.

Best-first search through the entire space of potential explanations, however, has some untoward effects on system users. In particular, it leads to "jumping around" rather than a thorough evaluation of particular solution paths. "Localized best-first" search corrects this. On each evaluation cycle, TEST+ gives preference to the potential explanations for its focal failure. It evaluates all of these and pursues the most likely explanation that is over threshold. It is only when none of the alternatives is above threshold that TEST+ refocuses on the globally best candidate. This approach was easier for end users to understand and resulted in greater acceptance of the system.

4.3 User Control

TEST+ allows users to inspect state information and to control diagnostic behavior in order to support diagnostic decisions. By making users aware of all conclusions that are over threshold, TEST+ enables decisions on whether or not to halt operations for repair. By allowing users to adjust the GCT, TEST+ lets them balance off thoroughness against the need for a quick response. The lower this threshold the more exhaustive the search. Finally, with the ability to select the failure mode focus, users can get TEST+ to assist them in pursuing their best hunches.

5 Evaluation of TEST+

In order to evaluate TEST+, expert diagnosticians were asked to assess TEST+'s effectiveness on hypothetical and real cases, as well as to compare TEST+ to TEST. TEST+ performed accurately on both hypothetical and real problems. In one "live" test, where the system ran in parallel with an expert, TEST+ diagnosed a cracked rotor – a problem which can result in million dollar damage if not identified early. Like the expert, TEST+ correctly identified the fault and had appropriately "ruled out" other possible faults. In light of several successful tests, turbine-generator technicians found

TEST+ outperformed themselves by more quickly disregarding weakly supported conclusions.

A TEST turbine-generator knowledge base was developed for purposes of comparison. Although the TEST knowledge base supported efficient diagnosis in some cases, it was found to be less accurate and flexible than the equivalent TEST+ system. TEST performed well where there was sufficient evidence to confirm failures, but not well under conditions of uncertainty. Technicians were frustrated by their inability to interact with the TEST search process, as well as by TEST's inability to assess the likelihood of candidate conclusions it had disregarded. The technicians considered these critical features of TEST+.

In evaluating TEST+, ideas emerged for potential improvements. For one, it appears desirable to focus immediately on failure modes highly implicated by the evidence, rather than to fully evaluate a set of candidates. This approach would have provided more efficient diagnosis across several test cases.

A second enhancement is to allow a certainty threshold to be defined locally for each failure mode. Among other things, this would enable failure mode selection to be more sensitive to the cost of ignorance – that is, lower thresholds would be assigned to problems that result in more costly losses.

Both these enhancements, however, have potential costs. In the first case, one could expect failures to go undetected; in the second, there would be increased complexity in the definition, maintenance, and validation of the knowledge base. Further investigation is required to appropriately assess the tradeoffs.

6 Conclusions and Future Directions

Diagnostic tasks vary with respect to the certainty of available data, degree of access to component failures, conclusiveness of test results, and diagnostic complexity. TEST+ provides a variant of the TEST architecture to address reasoning under uncertainty and diagnostic complexity. Integrating the strengths of inexact, evidential systems with the model-based approach of TEST, TEST+ provides a unique and powerful solution to the diagnosis of turbine-generators.

Understanding why TEST+ applies better than TEST to the turbine-generator domain enables us to conjecture that TEST+ may be appropriate to other domains with similar characteristics. The domains of medicine, nuclear reactors, and jet engines, among others, appear to have the critical characteristics necessary for the application of TEST+. More specifically, all of these domains share the characteristics of uncertainty in available data, inaccessibility to component failures, inconclusiveness of failure hypotheses, and highly complex failure states. Nevertheless, caution is advised. As the comparison of TEST and TEST+ shows, "domain-independent" shells have implicit assumptions about domain characteristics built in.

7 Acknowledgements

Acknowledgement and gratitude is due to Peter Sizer, Graham Hill, Rob Corlett, and Robin Khan of GEC Research, England for their contribution to our ideas as well as for knowledge acquisition; Claude LePape and Anne Collinot for their review and criticism; and especially to Donna Herbert, Carnegie Group, for her support and suggestions.

8 References

[1] Clancey, W., "Heuristic Classification," **Artificial Intelligence**, Vol. 27, No. 3 (1985), pp. 289-350.

[2] Fox, M. S., Kleinosky, P., and Lowenfield, S., "Techniques for Sensor-Based Diagnosis," **Proceedings of the Eighth International Joint Conference on Artificial Intelligence** (1983).

[3] Hayes-Roth, F., Waterman, D. A., and Lenat, D. B., **Building Expert Systems**, Addison-Wesley, Reading, Mass. (1983).

[4] Kahn, G. S., "On When Diagnostic Systems Want To Do Without Causal Knowledge," **Proceedings of ECAI'84: Advances in Artificial Intelligence** (1984).

[5] Kahn, G. S., Kepner, A., and Pepper, J., "TEST: A Model-Driven Application Shell," **Proceedings AAAI-87**, William Kaufmann, Los Altos, Ca. (1987), pp. 814-818.

[6] McDermott, J., "Preliminary Steps Toward a Taxonomy of Problem-Solving Methods," in **Automating Knowledge Acquisition for Expert Systems**, Marcus, S., ed., Kluwer Academic Publishers (1988), pp. 225-256.

[7] Seligman, L., Kahn, G., Pepper, J., and Kepner, A., "TEST: An Architecture for Troubleshooting Systems," **Proceedings of AI-87**, Long Beach, Ca. (1987).

[8] Shortliffe, E., **Computer-Based Medical Consultation: Mycin**, Elsevier (1976).

[9] Shrobe, Howard, editor, **Exploring Artificial Intelligence**, Morgan Kauffman (1988).