A REASONING STRATEGY FOR FAULT DIAGNOSIS

Won Young Lee,Ph.D.
Department of Industrial Engineering
Seoul National Polytechnic University
172 Kongnung-Dong, Nowon-Gu
Seoul, KOREA 139-743

요약 - 본 논문은 어떤 시스템 (예를 들면, 자동화된 제조시스템)에서 발생하는 징후에 대한 고장진단 모델을 개발하는 것이 목적이다. 이 모델은 계층적 시스템이론(Theory of Hierarchical Systems) 과 인공지능의 혼성추론 기법 (Hybrid Reasoning Approach) 을 사용한다. 일반적으로, 시스템은 스트라타 (strata) 와 에셜론 (echelons) 으로써 표현될 수 있으며, 한편 시스템에 대한 지식은 근본지식 (deep knowledge) 과 경험지식 (shallow knowledge) 으로 나뉘어 질 수 있다. 이 모델에서의 고장진단에 대한 추론전략은 근본지식베이스에 의한 근본적 추론을 먼저하고 그 다음에 경험지식베이스에 의한 경험적 추론을 하는 혼성추론 기법이다.

I. INTRODUCTION

It is clear that, as the scale of a system becomes larger (say, more than one hundred components), system maintenance becomes very difficult. If any component in the system is defective, this component should be found and fixed quickly. However it is difficult for one person to manage the whole diagnostic task for a large-scale system. In order to make the system run smoothly with minimum delay, it is necessary to have a diagnostic system for discovering any cause of system failure. Thereby, there is a need for intelligent diagnostic systems, which motivates this research.

In AI and expert systems, diagnosis has been given more attention in recent years, which includes trouble-shooting in electronic circuits (deKleer and Williams [1987]) and medical diagnosis (Gordon and Shortlffe[1985]).

In general, a symptom is a fact that a certain function does not occur. Thus a symptom is observed when the system behaves in a way that is not expected. In other words, a symptom is the discrepancy between the observed (or actual) behavior and the expected (or predicted) behavior of the system. Hence, when a symptom exists, there must be a fault(s) which cause(s) the symptom.

A diagnosis is the process of finding the location of the fault (i.e., which component of a syystem is faulty). The diagnosis can also suggest the remedy for the fault found. The diagnostic process can be summarized as fault detection, fault localization, and fault isolation (Khaksari[1988]). Fault detection consists of observing a symptom when a function of the system does not work properly. Sensors might indicate the presence of the symptom. Fault localization is a process of filtering out possible causes until we can focus on a specifically identifiable and feasible cause. Fault isolation is a search procedure for determining the source of the symptom. In words, a diagnosis procedure is a search procedure for finding faults, or goals.

Ⅱ. REVIEW OF REASONING METHODOLOGY

The fundamental task of diagnosis is the establishment of the reasoning process, using the knowledge given. Reasoning is to infer something logically from the knowledge given. This knowledge can be either fundamental or experiential. In the context of diagnosis, three types of reasoning are dominant, namely, deep, shallow, and hybrid (Milne[1987]; Pan and Tenenbaum [1986]; Reiter[1987]). Two are distinctive and the other is a combination. This chapter reviews the reasoning approaches for diagnosis.

1. Deep Reasoning

Deep reasoning, based on deep knowledge, is often referred to as model-based reasoning (Hamscher and Patil[1989]) since it uses a model of the system as a basis for inference. A model itself is constructed from characteristic information on the structure and behavior of a system being diagnosed. That is, a model can be structural, behavioral, and functional.

Deep reasoning, relative to shallow reasoning which will be discussed next, is more flexible and thorough, but slower. "More flexible" means that since deep reasoning is not domain-specific, it is easier to modify the model when a single component is added or deleted. Deep reasoning may not be sensitive to the change. "More thorough" means that deep reasoning may answer "what-if" type questions which may not be possible in shallow reasoning, as shown in Sembugamoorthy and Chandrasekaran[1986]. What this implies is that there is no limitation of fault coverage in deep reasoning. "Slower" means that the speed of reasoning is slower than that of shallow reasoning because a deep knowledge base does not contain every detail of a symptom.

2. Shallow Reasoning

Since shallow reasoning is highly domain-specific, diagnosis is fast if the symptom has beem experienced and thus has been included in the knowledge base. This reasoning typically uses (production) rules which consist of antecedents and consequents. An antecedent is a condition part and a consequent is an action part of a rule. If certain conditions are met, then some actions are performed. For this reason, such a rule is often called an IF-THEN rule. These rules can be classified by their behavior, i.e., self-managing rules and meta rules. A self-managing rule is one in which actions are performed without referring to any other rules. A meta rule is one in which its actions result in the triggering of other rules.

A problem, however, is that shallow reasoning is rigid in the sense that there may have to be substantial changes in the rules even with the simple addition or deletion of a single component. Consequently, the number of rules becomes practically unmanageable as the number of components of the system being diagnosed increases. If multiple faults can occur simultaneously, then this approach becomes combinatorially explosive.

3. Hybrid Reasoning

Shallow reasoning has been widely used but, because of the disadvantage mentioned above, deep reasoning has emerged. However, since it requiresmore search time and thus shows an undesirable speed of reasoning for some complex systems, deep reasoning alone is not satisfactory either. Hence the combination (called hybrid reasoning) of these two approaches has been attempted in order to perform the diagnostic process efficiently (from deep reasoning) and effectively (from shallow reasoning). In other words, hybrid reasoning utilizes both deep and shallow reasoning methodologies in an attempt to take advantage of the strengths of each. Two possible means of combination are (1) deep first, then shallow (D-S) and (2) shallow first, then deep (S-D).

Although there has been no comparative study for various types of reasoning, it is conceivable that, when the system to be diagnosed is relatively small (this also implies a small number of rules), an S-D approach seems to be preferred. For a large-scale system like a manufacturing plant, it seems that the D-S approach is often chosen.

III. REASONING STRATEGY

1. System Modeling

Mesarovic et al. [1972] claimed that most large-scale systems such as steel industries and electric power systems can be hierarchically organized. They also categorized hierarchical systems into three types, depending on three notions of levels. These three levels are: level of description or abstraction, level of decision complexity, and level of organization. These are termed as strata, layers, and echelons, respectively.

In general, strata are used for the purpose of modeling, layers are used for ease of vertical decomposition of a complicated decision-making problem, and echelons are used for representing mutual relationships berween decision-making units. However, as a method of representing a hierarchical system, we may adopt all three notions of levels. We are not restricted to use only one notion of level even though we may be forced to use only one notion of level from the analysis of the nature of a given system.

2. Structuring for Knowledge Bases

The model developed in this paper consists of a single Deep Knowledge Base (DKB) formed as strata and a number of Shallow Knowledge Bases (SKBs) formed as echelons and attached to each of the terminal nodes in the DKB. The DKB is constructed by viewing the whole system under consideration as a hierarchical system. The SKB is also organized hierarchically, based on the level of decision complexity.

1) Deep Knowledge Base

A system is represented by functional blocks within a deep KB in the form of tree (i.e., hierarchy) as shown in Figure 1. Even though the entire figure resembles a multiechelon system, each level is called a stratum, not an echelon, because the figure represents levels of (functional) description or abstraction of the system. In the deep KB,a "functional block" is referred to as a "node." No test (about whether a functional block works properly) is associated with every node in the deep KB and, in turn, "no test" implies "no decision-making." For this reason, every node is connectea downwards by a one-way arrow.

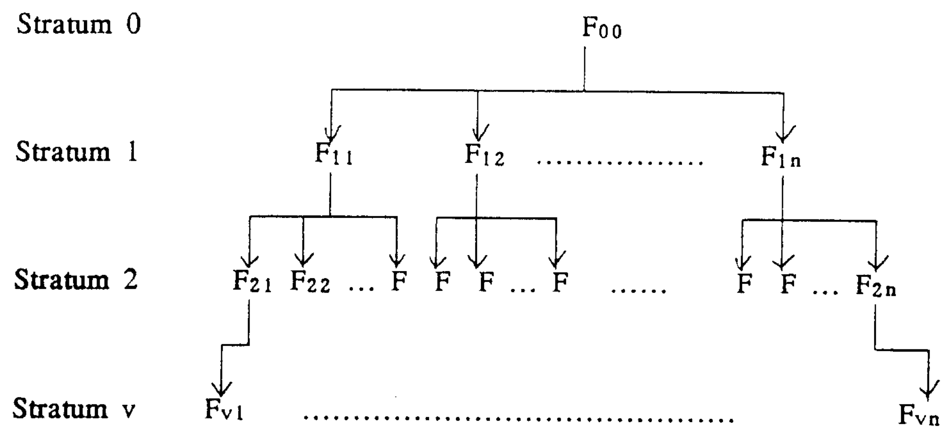

Figure 1. Functional hierarchy as deep knowledge base

Except for F_{00} which represents the given system as a whole, it is assumed that functional blocks are independent of each other within each stratum. Every terminal node has its own shallow KB.

With the exception of F_{00}, a functional block F_{ij} is defined as follows:

i indicates the stratum number, $i = 1,2,\cdots,v$

j indicates the "overall" element number, $j = 1,2, \cdots ,n$

where v and n are arbitrary integers.

This definition of a functional block is shown in Figure 1.

2) Shallow Knowledge Base

A shallow KB, attached to each of the terminal nodes in the deep KB, is organized in multiechelon form because every rule involves its own decision-making. This decides whether or not the rule is responsible for the observed symptom after testing the consequent of the rule. For this reason, every rule is connected by a two-way arrow. Each level is called an echelon, as shown in Figure 2.

Unlike F_{ij}, a rule R_{ijk} is defined as follows:

i indicates echelon i, $i = 1,2, \cdots , e$

j indicates group j, $j=1,2, \cdots$, up to s

where s is the total number of elements in Echelon i-1

k indicates element k,$k = 1,2,\cdots$, up to t

where e, s, and t are arbitrary integers.

In the shallow KB, two attributes are associated with every rule R_{ijk} at and below Echelon 3 (i.e., R_{ijk} where $i \geq 3$, and j and k are arbitrary). One is p_{ijk}, defined as a degree of belief, e.g., obtained from the opinions of experts, which acts as a probability that R_{ijk} is believed to be responsible for the obesrved symptom. Furthermore, p_{ijk} is designed to have the following property:

$$\sum_{k=1}^{t} p_{ijk} = 1$$

where $i \geq 3$ and j is arbitrary.

In other words, at and below Echelon 3, the sum of the degrees of belief of all the elements within any arbitrary jth group is unity. This property provides ease of coding in Lisp. The other is c_{ijk}, defined as a test cost (in dollars), which accounts for the cost of testing the consequent part of R_{ijk}.

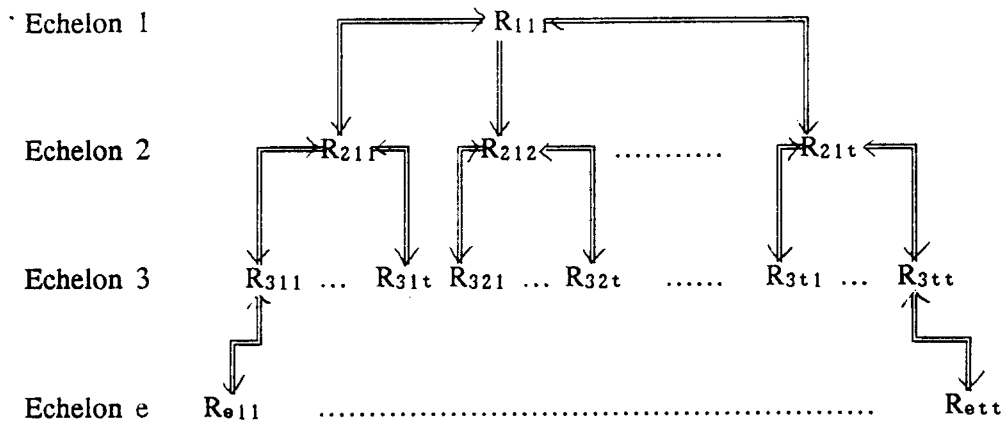

Figure 2. Rule hierarchy as shallow knowledge base

3. Diagnostic Reasoning Strategy

The diagnostic reasoning strategy is an extension of the strategy by Lee et al.[1990]. The strategy requires the failure probabilities of functional blocks for the deep KB, the degrees of belief, and the associated test costs of rules for the shallow KB. Also we will assume the followings:

a. A symptom is directly observable for the system under consideration.
b. For the duration of the diagnostic process, all components in the system maintain their status.
c. The terminal nodes (i.e., rules) in the shallow KB are mutually exclusive (i.e.,disjoint) and exhaustive.

1) Outline

Based on the assumptions above and given the failure probabilities for the deep KB and the degrees of belief and test costs for the shallow KB, this model identifies and isolates a fault by searching the deep KB first and then the shallow KB. Backtracking is allowed if needed. The outline of the diagnostic strategy is illustrated in Figure 3. Given a single symptom, the basic cycle of the strategy is to find out a single fault for this symptom. The basic cycle consists of 8 steps which are described below. Details of these 8 steps will be provided in the next section.

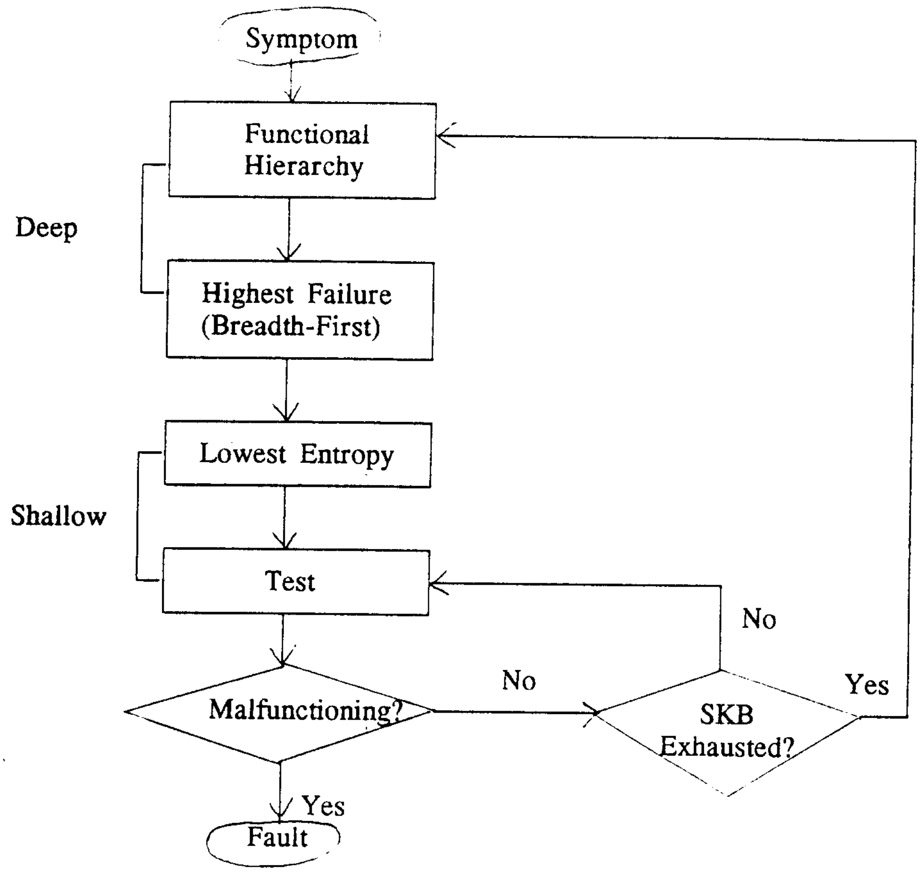

Figure 3. Schematic diagram for diagnostic strategy

A single symptom means that we observe a symptom only once and determine which component(s) is (are) faulty. Multiple faults for the symptom observed can be handled by testing the possible faulty components one by one, sequentially. That is, multiple faults are found by continuing the current diagnostic session even if one fault has already been found. Multiple symptoms can also be treated similarly. However, it will be much better if we can classify these multiple symptoms, based on functional levels. The reason is that the further we go down in the functional hierarchy, the more we reduce the search time required to find the fault.

The basic cycle of 8 steps embedded in Figure 3 are as follows:

Step 1. Construct a functional hierarchy of the given system.

Step 2. When a symptom is observed, go to the deep KB and start at the root F_{00}.

Step 3. Conduct a breadth-first search using the responsibility probability based on the failure probability of each of the functional blocks at the immediate lower stratum. Select the most unreliable branch.

Step 4. Repeat Step 3 with the functional block selected until the terminal node is reached. Then go to the shallow KB attached to the functional block finally found.

Step 5. Use entropy to determine the rule yielding the lowest entropy. Repeat the entropy calculations until the terminal node is reached.

Step 6. Test the "consequent" part of the rule(i.e., the terminal node) found in the previous step. If the consequent is malfunctioning, then stop andassume it is faulty. If it is functioning, then continue the search.

Step 7. Back up to the immediate higher echelon and select the rule having the next higher entropy. Repeat Step 6.

Step 8. If the right rule is not found from the shallow KB, then go back to the terminal node of the deep KB and update the responsibility probabilities. Starting at the root F_{00}, conduct the same breadth-first search until theterminal node is reached. Then go to Step 5.

2) Details

Steps 1 through 4 are concerned with deep reasoning and are self-explanatory.

Steps 5 through 7 are associated with shallow reasoning. Here we explain only Steps 5 and 6 since Steps 7 and 8 are self-explanatory.

Step 5

For the shallow KB, we use the concept of entropy (Zeleny[1982]) in determining the responsible rule.

The traditional Shannon's entropy (Zeleny [1982]) is defined as:

$$H(p) = -\sum_{i=1}^{n} p_i \ln p_i \quad ,$$

where n is the number of states of nature with estimated probabilities $p_1,\cdots,p_n$

That is, H(p) represents the expected amount of information contained in a given information source.

Shannon's entropy, by definition, does not incorporate any other attributes like test cost but estimated probabilities. In order to accommodate other attributes, Shannon's entropy must be modified. This modified form of entropy is called "useful" information (Sharma et al.[1978]) on which our entropic measure is based.

Referring to the shallow KB in Figure 2, we start making a decision at RlII to select the rule to be tested nest. What we need at this point are entropies of the rules at Echelon 2. This, in turm, means that we need to know the probabilities (degrees of belief) and the test costs at Echelon 3. This procedure is repeated until the terminal urle is reached.

Moreover, note that two units are noncommensurate, i.e., c_{ijk} is in dollars and p_{ijk} is in nondimensional. For this reason, c_{ijk} is normalized as follows:

$$w_{ijk} = \frac{c_{ijk}}{\max_{i,j,k} \{ c_{ijk} \}}$$

where w_{ijk} represents the weight of a rule R_{ijk}.

Then, given the degrees of belief and the test costs, the entropy of a rule R_{ijk} is defined as follows:

$$H_{ijk} = H_{ijk}(w,p) = -\sum_{x=1}^{t} w_{i+1,k,x}\, p_{i+1,k,x} \ln p_{i+1,k,x} \ , \quad i \geq 2$$

where $w_{i+1} \geq 0, \sum_{x=1}^{t} p_{i+1,k,x} = 1.$

Here j just indicates the group number from the immediate higher echelon and x denotes the element k from Echelon i+1. When $w_{ijk} = 1$ for all i,j, and k (i.e., all test coasts are equal), our entropy equation reduces to Shannon's entropy. Notice that all the distributed or assgined beliefs within a group sum to unity.

In short, we would like to determine the rule which has the lowest entropy. One remark here is that when a tie occurs (i.e., there are two or more rules that have the same entropies) at the terminal rules whoses entropies are zero, the ratio, weight/probability, is computed in order to break the tie. This ratio is defined as

$$r_{ijk} = \frac{w_{ijk}}{p_{ijk}}$$

So, the tie is broken by selecting the lowest ratio which gives the lowest test cost. A tie at other rules can be broken "arbitrarily" since these rules do not directly involve the testing (i.e., the entropy of each of these rules is not zero).

The overall feature of the search in the shallow KB is similar to the best-first search because the search is performed in increasing order of entropy.

Step 6

When the terminal node is reached, it is tested to determine whether or not the component in the consequent part of the rule is functioning properly. If the component is not working properly after the test, it is deemed fautly and must be fixed. The diagnostic session stops at this point. If it is working after the test, then proceed to the next step (i.e, Step 7). If, however, the system still has the same symptom as before after getting the suspected compenent fixed, the symptom may have multiple simultaneous faults. Thus the diagnostic session must be continued (i.e, Step 7 has to be performed).

IV. CONCLUSION

We outlined a reasoning strategy for fault diagnesis. However some of the detaile have been curtailed. Currently we are experimenting whether a diagnosis model using this strategy would be applicable to any domain.

REFERENCES

de Kleer, J.and B.C.Williams, "Diagnosing Multiple Faults, "Artificial Intelligence, Vol.32, No.1, Apr.1987, pp 97-130

Gordon, J.and E.H.Shortliffe, "A Method for Managing Evidential Reasoning in a Hierarchical Hypothesis Space," Artificial Intelligence, Vol.26,No.3, July 1985, pp 323-357

Hamscher, W.and R.Patil, Model-Based Diagnosis:Tutorial, at the Eleventh Int'l Joint Conf.on Artificial Intelligence (IJCAI-89), Detroit, MI, Aug.21, 1989

Khaksari, G.H., "Expert Diagnostic System, "Proc.of the First Int'l Conf. on

Industrial & Engineering Applications of AI and Expert Systems '88, Vol.I,June 1-3, 1988, pp 43-53

Lee, W.Y., S.M.Alexander, and J.H.Graham, "A Hybrid Approach to a Generic Diagnosis Model,"Proc.of the Fourth Int'l Conf.on Expert Systems in Production and Operations Management, Hilton Head Island, SC, May 14-16, 1990,pp 264-273

Mesarovic, M.D., D.Macko, and Y.Takahara, Theory of Hierarchical, Multilevel Systems, Academic Press, New York, 1972

Milne, R., "Strategies for Diagnosis, "IEEE Trans.on Systems.Man.and Cybernetics, Vol.SMC-17, No.3, May/June 1987, pp 333-339

Pan, J.Y.C. and J.M.Tenenbaum. "PIES:An Engineer's Do-It-Yourself Knowledge System for Interpretation of Parametric Test Data," AI Magazine, Vol.7, No.4, Fall 1986, pp 62-69

Reiter, R., "A Theory of Diagnosis from First Principles, "Artificial Intelligence, Vol.32, No.1, Apr. 1987, pp 57-95

Sembugamoorthy, V.and B.Chandrasekaran, "Functional Representation of Devices and Compilation of Diagnostic Problem-Solving Systems," in Experience, Memory, and Reasoning, edited by J.L. Kolodner and C.K.Riesbeck, Erlbaum Inc., Hillside, NJ, 1986, PP 47-73

Sharma, B.D., J.Mitter, and M.Mohan, "Our Measures of 'Useful' Information." Information and Control, Vol.39, No.3, Dec.1978,pp 323-336

Zeleny, M., Multiple Criteria Decision Making, McGraw-Hill, NY,1982,Chapter 7