Robotica (2001) volume 19, pp. 669–674. Printed in the United Kingdom © 2001 Cambridge University Press

An Expert System Shell for Fault Diagnosis*

Jie Yang, Chenzhou Ye and Xiaoli Zhang

Institute of Image Processing & Recognition, Shanghai Jiao-Tong University, Shanghai 200030 (P. R. of China)

(Received in Final Form: March 20, 2001)

SUMMARY

Traditional expert systems for fault diagnosis have a bottleneck in knowledge acquisition, and have limitations in knowledge representation and reasoning. A new expert system shell for fault diagnosis is presented in this paper to develop multiple knowledge models (object model, rules, neural network, case-base and diagnose models) hierarchically based on multiple knowledge. The structure of the expert system shell and the knowledge representation of multiple models are described. Diagnostic algorithms are presented for automatic modeling and hierarchical reasoning. It will be shown that the expert system shell is very effective in building diagnostic expert systems.

KEYWORDS: Expert system shell; Fault diagnosis; Rough set; Artificial Neural Network; Case-based reasoning.

1. INTRODUCTION

Fault Diagnosis is a very important area of research in Artificial Intelligence. Existing diagnostic approaches can be grouped into four classes: Model-based diagnosis, rule-based diagnosis, neural network-based diagnosis, case-based diagnosis. The knowledge representation and reasoning of model-based and rule-based diagnoses are symbol-based, while those of neural network-based and case-based diagnoses are instance-based. Model-based diagnosis[1,2] uses an operate model that describes the operating behavior of the device to be examined. Rule-based diagnosis uses rules in abductive reasoning, representing empirical associations that relate symptoms to the underlying malfunctions. Instance-based diagnosis uses instances of diagnosis accumulated by experience; it utilizes the similarity between the current situation and previously known instances to come up with a plausible diagnosis. For the diagnostic approach using neural network, responses to the inputs are determined by the similarity between the input and the prototypical instances in previous training data, which can be considered as performing feature clustering based on similarity or as performing statistical inferences. To diagnose a device, case-based diagnosis searches a case-base to find a case that best matches the current situation. The diagnosis and solution of the case will be adapted for the diagnosis of the device. The approaches of these four classes use different models for knowledge representation, modeling and reasoning. Each of them has corresponding advantages and can complement each other.[3,4] But existing diagnostic expert systems are seldom used to represent multiple knowledge models and diagnoses, based on multiple knowledge models. Usually, they cannot deal with nonmonotonic reasoning (exceptions in knowledge representation and reasoning).

As the building of an expert system for fault diagnosis is generally a complicated and time-consuming task, an expert shell is very useful and efficient. Traditional expert systems for fault diagnosis have a bottleneck in knowledge acquisition. For many complicated objects (e.g. chemical equipment), the operating and diagnostic models of the objects are difficult to build directly by experts or simulation. Traditional expert system shell cannot deal with knowledge acquisition, knowledge representation and combined reasoning of multiple models We present a new expert system shell which can efficiently represent and automatically model multiple diagnostic knowledge. It diagnoses objects by hierarchical reasoning based on multiple models. The structure of the new expert system shell is described in Fig. 1.

In the expert shell, a instance-base of operating behavior and a instance-base of fault behavior are built at first. Multiple knowledge models are acquired by a data mining and learning mechanism[5] from two bases. According to the fault behavior of an object examined, the blackboard-based inference mechanism realizes diagnostic reasoning hierarchically by multiple models in a knowledge-base. After a result of the diagnosis is inferred, consistency of the result is maintained by truth maintenance. The explanation of the result is also given in association with the knowledge models used in diagnosis and consistency examination. The knowledge-base is then revised by learning from the failures in diagnostic processes.

2. KNOWLEDGE REPRESENTATION

In object diagnosis, when the instance-base of operating behavior and the instance-base of fault behavior are built, four knowledge models (operate model, rule-base, neural model, case-base) are acquired by the data mining and learning mechanism and represented in the knowledge-base in the order from general to specific.

Knowledge at the top level is composed of an operation model describing the most general diagnostic knowledge. For rule-based or case-based diagnosis, abnormal symptoms need to be determined according to the behavioral discrepancy between predicated behavior and observed behavior. The operation model, which is represented by fuzzy rules to realize input-output data approximation, is used to compute

* This research is supported by National Hi-Tech Plan "863-CIMS" and National Science Foundation of China.

https://doi.org/10.1017/S0263574701003460 Published online by Cambridge University Press

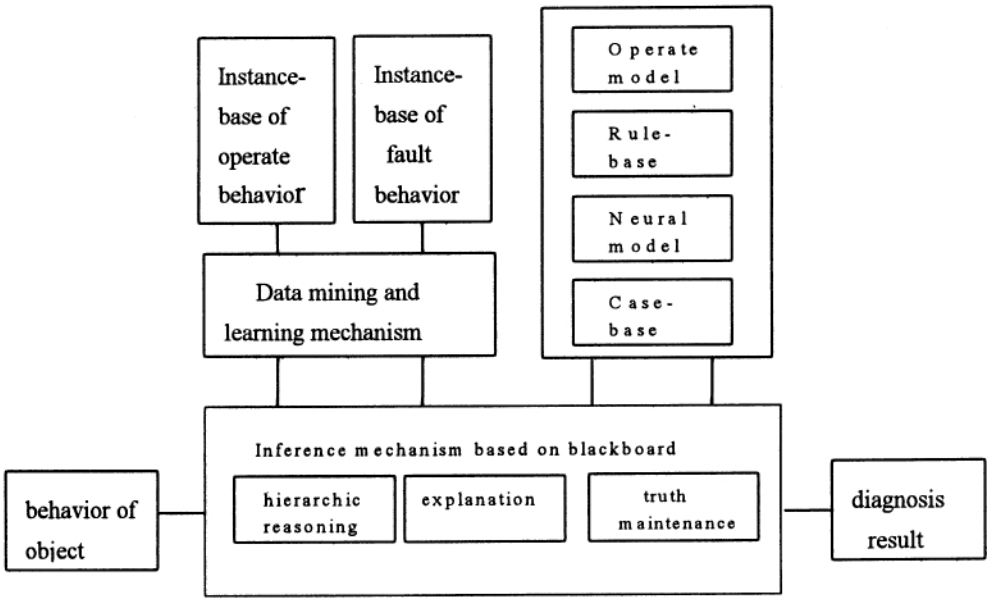


Fig. 1. The structure of an expert system shell for fault diagnosis.

predicated state features and realize truth maintenance in fault diagnosis. The form of a fuzzy rule is: if $I_1=A_1, \ldots,$ $I_n=A_n$ then $O_1=B_1, \ldots, O_m=B_m$, where $I_1, \ldots, I_n$, $O_1, \ldots, O_m$ are input variables and state feature variables of the object to be diagnosed, $A_1, \ldots, A_n$, $B_1, \ldots B_m$ are the categories of fuzziness of the corresponding variables. Triangular membership functions are used to define the categories of fuzziness of each variable.

Knowledge at the second level consists of diagnostic rules that describe general diagnostic knowledge and are used for rule-based abductive diagnosis. The form of a diagnostic rule: if $X_1, \ldots, X_n$ then Y represents associations that relate symptoms $X_1, \ldots, X_n$ to the underlying malfunction *Y*.

Knowledge at the third level is built from neural models that describe relatively specific diagnostic knowledge and are used for neural network-based diagnosis.

A multi-layer perceptron network is selected for the construction of neural models. Nodes $Z_1, \ldots, Z_M$ in the input layer represent the descriptors of symptoms of an object diagnosed. The nodes $O_1 \ldots O_K$ in the output layer represent the result of diagnosis (Fig. 2).

Knowledge at the bottom level is composed of cases that describe the most specific diagnostic knowledge. Cases are represented as frames. A feature of a case is a default representation of a symptom in the case. Features descriptors characterize the value or value domain for features, and are used for case classification. Each case is given a string of binary signs to indicate the status of its features. The sign of a feature is 1 if the feature is abnormal, and 0 if normal. The signs of features are used for case organization and case search. Each case representation also gives its diagnosis and the constraints among features.

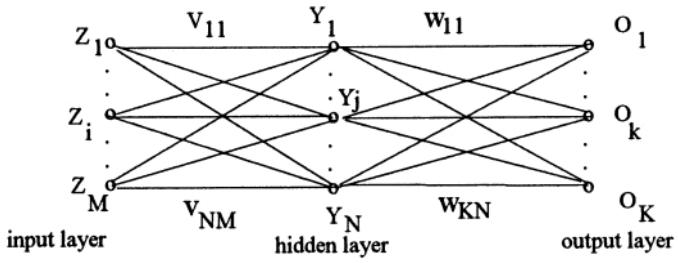


Fig. 2. Multi-layer perceptron network.

(*C*1 (signs 1,0, · · ·) (featurel *F*1) (feature2 *F*2) . . .

· · ·(constraint *CON*1) (diagnosis *S*1))

The above frame represents case *C1*: featurel is abnormal and its value is *F1*; feature2 is normal and its value is *F*2; the features in *C1* should satisfy the constraint *CON1*; the diagnosis of *C1* is *S1*.

As the knowledge model at a lower level provides more specific diagnostic knowledge than that at a higher level, the knowledge model at a lower level may represent exceptions of that at a higher level.

3. AUTOMATIC MODELING FOR FAULT DIAGNOSIS

3.1 Learning fuzzy rules as operate model by genetic algorithm

When the instance-base of operating behavior is known, each variable is divided into several categories of fuzziness whose membership functions are defined as triangular functions. To operate a model of an object diagnosed, fuzzy rules[6–7] of automatic learning can be considered as the problem of combination optimization of input-output states; that is, a group of fuzzy rules can be treated as a kind of combination. Fuzzy rules of automatic learning are to find the best one. As an effective technique of optimization, a genetic algorithm (GA) can be used for automatic learning fuzzy rules.[8] Key techniques in automatic learning fuzzy rules by GA are chromosome coding of fuzzy rules, operators of inheritance (selection, crossover, mutation), and evaluation of chromosome (fuzzy rules).

https://doi.org/10.1017/S0263574701003460 Published online by Cambridge University Press

Chromosome coding of fuzzy rules. A group of fuzzy rules can be considered as a kind of combination of fuzzy predicates. As the order of the combinations of fuzzy predicates of input variables can be determined in advance, only fuzzy predicates of output variables (conclusions of the fuzzy rules) under these combinations need be recorded for chromosome coding of fuzzy rules. As the number of categories of fuzziness of output variable is usually not more than 9, numbers between 1 and m (≤9) are used to represent the categories of fuzziness of output variable (m is number of the categories of fuzziness of output variable).

I_1	S	S	S	S	S	S	S	S	S	M	M	M	M	M	M	M	M	M	L	L	L	L	L	L	L	L	L	L
I_2	S	S	S	M	M	M	L	L	L	S	S	S	M	M	M	L	L	L	S	S	S	M	M	M	L	L	L	
I_3	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	
O_1	L	L	M	M	S	M	L	L	M	M	M	S	M	L	L	M	S	M	L	M	M	S	S	M	L	M	S	

For example, chromosome coding of above fuzzy rules with three inputs and one output are realized by coding fuzzy predicates of O_1: 332212332221233212322112321 (1, 2, 3 represent the fuzzy predicates S, M, L respectively). On the contrary, given a chromosome coding of fuzzy rules 112233221233221232211122322, it can be transformed into corresponding fuzzy rules as follows:

I_1	S	S	S	S	S	S	S	S	S	M	M	M	M	M	M	M	M	M	L	L	L	L	L	L	L	L	L	L
I_2	S	S	S	M	M	M	L	L	L	S	S	S	M	M	M	L	L	L	S	S	S	M	M	M	L	L	L	
I_3	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	
O_1	S	S	M	M	L	L	M	M	S	M	L	L	M	M	S	M	L	M	M	S	S	S	M	M	T	M	M	

Operations of chromosome inheritance. (1) Selection: The random selection of chromosome for reproduction is based on a biased roulette wheel algorithm. (2) Crossover: When two chromosomes are randomly selected for the operation of crossover, two points are randomly selected in the length of these two chromosomes. Two new chromosomes are generated by performing a two-point crossover. (3) Mutation: Given a chromosome for the mutation operation, one point P is randomly selected in the length of the chromosome, the allele of the chromosome at point P is altered randomly into n' which satisfies the constraint: $n' \leq m$, $n' \neq n$.

Evaluation of a chromosome (fuzzy rules). Given the instance-base of operating behavior about inputs and their expected outputs, the goal of learning fuzzy rules is to reduce the error between expected outputs and outputs derived by an operate model based on fuzzy rules as small as possible. So evaluating the fitness of a group of fuzzy rules can be realized by computing the cumulative errors between expected outputs and outputs derived by an operate model under training examples in the instance-base of operating behavior. Given a chromosome operated by crossover and mutation, it can be transformed into its corresponding fuzzy rules. According to inputs of training examples, outputs of the operating model can be derived by predefined algorithms (detailed in section 4): fuzzification, MIN-MAX fuzzy inference according to the fuzzy rules learned by GA, defuzzification. The fitness of the group of fuzzy rules is evaluated by computing cumulative errors between real outputs and expected outputs of training examples.

$$f = \sum_{i=1}^{n} \sum_{j=1}^{m} (O_j - O_{ej})^2$$

where n – the number of training examples; m – the number of output variables; O_j – the output derived by fuzzy rules and inputs of training example T_i; O_{ej} – the expected output of training example T_i.

The following example is used to examine the algorithm. An object with three inputs and one output is constructed, its mathematical model is:

$$o = 0.5 \times (e^{i_1} + i_2 \times i_3 \times \cos(i_1 \times i_2) + i_1 \times i_3),$$

where $i_1, i_2, i_3, o \in [0,1]$. A Fuzzy system is learned for the modeling of the system by our developmental tool, i_1, i_2, i_3 are divided into 5 categories of fuzziness. In order to show the performance of the fuzzy system in graphic form, i_1, i_2, i_3 are represented by functions with a same variable $t \in [0,1]$: $i_1 = t$,

$$i_2 = \begin{cases} 2 \times t & 0 \leq t < 0.5 \\ 2 - 2 \times t & 0.5 \leq t \leq 1 \end{cases},$$

$$i_3 = \begin{cases} 0.9 & t \in [0, 0.25) \cup [0.5, 0.75] \\ 0.1 & t \in [0.25, 0.5) \cup [0.75, 1] \end{cases}.$$

The following Fig. 3 is the result of the fuzzy system learned by genetic algorithm, the curve and the broken line in Fig. 3 separately represent the real output and expected output of the fuzzy system.

3.2 Learning diagnostic rules by rough set

The rough set[9,10] has recently emerged as a major mathematical approach for knowledge discovery and data mining, it is used as a tool in the expert shell for extraction of diagnostic rules from the instance-base of fault behavior.[11] Some preliminaries of rough set theory are presented as follows:

An Information system is a pair $S = \{U, A\}$, where U is a nonempty finite set called the universe and A a nonempty finite set of attributes. An attribute a can be regarded as a function from the domain U to some value set V_a. With

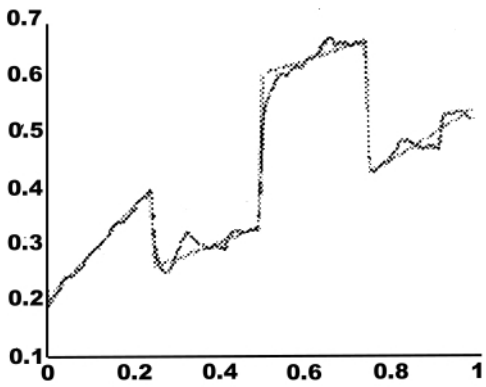


Fig. 3. The result of the fuzzy rules learnt by GA.

https://doi.org/10.1017/S0263574701003460 Published online by Cambridge University Press

every subset of attributes $B \subseteq A$, an equivalence relation I_B on U can be associated as: $I_B = \{(x, y) \in U$: for every $a \in B$, $a(x) = a(y))\}$ then $I_B = \cap_{a \in B} I_a$.

If $X \subseteq U$, the sets $\{x \in U: [x]_B \subseteq X\}$ and $\{x \in U: [x]_B \cap X \neq \Phi\}$, where $[x]_B$ denotes the equivalence class of the object $x \in U$ relative to I_B, are called the *B-lower* and *B-upper approximation* of X in S and denoted $\underline{BX}$, $\overline{BX}$, respectively.

The aim of knowledge reduction is to obtain irreducile but essential parts of the knowledge encoded by a given information system. Diagnostic rules are extracted according to the *Reduct*.[12]

The algorithm for heuristic *Reduct* Searching:

Input: Instance-base U, Threshold of correctness β, Attribute set $C = \{a_1, a_2, \ldots, a_n\}$, Fault Y, Weight set of attributes $W = \{w_1, w_2, \ldots, w_n\}$
Output *Reduct*
Methods: Construct $C^+ = \{X_1^+, X_2^+, \ldots, X_s^+\}$ and $C^- = C^* - C^+ = \{X_1^-, X_2^-, \ldots, X_t^-\}$
Calculate discernible set $A = \{A_1, A_2, \ldots, A_q\}$
Set Re*duct* $= \phi$, $R = \{a_i | \forall a_i \in A, i = 1, \ldots, n\}$
While $A \neq \phi$ Do {
For each a_l, calculate the number of appearance N_l in A.
For every $a_i \in R$ Do
If $\exists A_j = \{a_i\}$ Then the fact of a_l is: $Q_l = -1$
Else the factor of a_l is $Q_l = w_l * N_l$
Select the attribute a_k where $Q_k = \min\{Q_i | Q_i > 0 \quad i = 1 \ldots n\}$
Re*duct* $=$ Re*duct* $\cup \{a_i | Q_i = -1\}$
$A = A - \{A_j | \exists a_i \in$ Re*duct* $\cup \{a_k\}, a_i \in A_j\}$
If $N_k = |A|$ Then Re*duct* $=$ Re*duct* $\cup \{a_k\}$
}

The algorithm for extracting diagnostic rules is:

Input: β, C, Y, W
Output: *Diagnostic rule*
Methods: The *Reduct* of C is calculated according to the above algorithm.
Calculate Re$duct^+ = \{X_1^+, X_2^+, \ldots, X_s^+\}$ and Re$duct^- = \{X_1^-, X_2^-, \ldots, X_t^-\}$
For X_i^+, extract a diagnostic rule $DES(X_i^+) \xrightarrow{P(Y|X_i^+)} Y(Sup(Y|X_i^+))$, $DES(X_i^+)$ includes attributes only in *Reduct*. $P(Y|X_i^+) = |Y \cap X_i^+| / |X_i^+|$ is the conditional probability of Y on X_i^+, which represents the correctness rate of the classification Y on X_i^+; $Sup(Y|X_i^+) = |Y \cap X_i^+| / |Y|$ represents the support rate of Y on X_i^+.
For X_i^-, extract a diagnostic rule $DES(X_i^-) \xrightarrow{P(Y|X_i^-)} \neg Y(Sup(\neg Y|X_i^-))$, $DES(X_i^-)$ includes attributes only in *Reduct*. $P(Y|X_i^-) = |\neg Y \cap X_i^-| / |X_i^-|$ is the conditional probability of $\neg Y$ on X_i^-, which represents the correctness rate of the classification $\neg Y$ on X_i^-; $Sup(\neg Y|X_i^-) = |\neg Y \cap X_i^-| / |\neg Y|$ represents the support rate of $\neg Y$ on X_i^-.

For example, $U = \{\text{inst1} \sim \text{inst13}\}$, $C = \{\text{a1,a2,a3,a4,a5}\}$, $D = \{d\}$, $Y = \{d = 1\}$, $\beta = 0.75$

	Inst1	Inst2	Inst3	Inst4	Inst5	Inst6	Inst7	Inst8	Inst9	Inst10	Inst11	Inst12	Inst13
a1	0	1	1	0	0	0	0	1	0	0	0	1	0
a2	0	0	0	0	1	0	1	0	0	1	0	0	0
a3	1	0	0	1	0	1	0	0	0	1	0	1	0
a4	0	1	1	0	0	0	0	1	1	1	1	1	1
a5	1	0	0	1	0	1	0	0	0	1	0	0	0
d	1	0	1	1	0	1	0	1	0	1	1	1	1

Two diagnostic rules are extracted as follows:

$$a1 = 0 \wedge a3 = 1 \xrightarrow{1} Y(4/9), \quad a1 = 1 \wedge a3 = 0 \xrightarrow{0.75} Y(3/9),$$

After diagnostic rules are acquired from the instance-base of fault behavior, the instances that can be satisfied by the diagnostic rules are delected from the instance-base.

3.3 Constructing neural models

Neural models based on multi-layer perceptron network are constructed according to the instance-base of fault behavior. Nodes $Z_1 \ldots Z_M$ in the input layer represent the descriptors of symptoms of training instances of fault behavior. The nodes $O_1 \ldots O_K$ in the output layer represent the diagnosis results of the training instances. $d_1 \ldots d_K$ represents the desired outputs of $O_1 \ldots O_K$; i.e., $d_1 \ldots d_K$ represent desired diagnosis results of the training instances. Instances in the instance-base about the diagnosis of faults in an object are used for the training of multiple perceptron network. According to the Error Back-Propagation Algorithm and the differences between the desired and actual neuron's response, the weights of the output layer and the hidden layer $W \leftarrow W + \eta \delta_o y^t$, $V \leftarrow V + \eta \delta_y z^t$ are adjusted until cumulative cycle error is less than E_{max}.

After neuron models have been constructed by the instance-base of fault behavior, the instances satisfied by the neural models are deleted from the instance-base.

3.4 Organizing the case-base

According to the instance-base of fault behavior, case-bases about diagnosis of instances in the instance-base can be organized. Each instance in the base represents a case. The predicted state features of the object being diagnosed are computed by the operating model and the signs of abnormal features are determined according to the difference between observed features and predicated features of the instance. Cases are organized in the case-base by binary trees according to the signs of abnormal features. For a case with features $f_1 \ldots f_m$, its signs of features are a binary string with length of m. Based on the binary string, cases are sorted into binary trees. Each case is located in a leaf of the tree, and the length from a leaf to the root is m. Each leaf is a certain classification. The possible classifications of cases in the binary trees is 2^m maximum.

4. DIAGNOSTIC REASONING BASED ON MULTIPLE MODELS

After the knowledge-base of multiple diagnosis model has been built, the expert system can be used for fault diagnosis. When the observed behavior of an object to be diagnosed is inputted into the expert system, it is described in the blackboard of the inference mechanism. The inference mechanism uses multiple models in the knowledge-base to realize diagnosis.[3,4,13,14] The following are the strategies used by the inference mechanism for diagnosis.

(a) For a complicated object diagnosed, according to the functions of components in the object, the object is divided into some functional groups (that is; some components are combined to constitute a functional group). For the object, two levels of knowledge models are built: a knowledge model for the entire object and a knowledge model for the functional groups. In the knowledge model for the entire object, a functional group is considered as a component. The knowledge model of a functional group describes only the diagnostic knowledge of the functional group. The inference mechanism diagnoses faults in the object hierarchically. First, the knowledge model of the entire object is used to find the malfunctioning groups whose knowledge model is used then to localize fault components.

(b) For diagnosis of an object, the inference mechanism realization can use diagnostic reasoning in the order from

https://doi.org/10.1017/S0263574701003460 Published online by Cambridge University Press

general to specific. The operate model is used at first to compute a predicated state of the object. Abnormal symptoms are determined according to the difference between observed state features and predicated ones. Then diagnostic rules are used for the rule-based diagnosis. If the result derived from rule-based diagnosis doesn't work, neural models are used for a second level diagnosis. If the neural network-based diagnosis fails, a case-base is used to realize a more specific diagnosis. As the knowledge model in lower level provides a more specific diagnostic knowledge than that at a higher level, the inference mechanism can deal with nonmonotonic reasoning in diagnosis.[2]

(c) After the result of diagnosis is inferred with multiple knowledge model, the inference mechanism realizes truth maintenance by examining the consistency between the operate model and the result of the diagnosis. The inference mechanism provides explanations of the diagnosis by showing the predicated behavior and the knowledge model used for diagnosis.

(d) If faults in an object cannot be diagnosed with multiple knowledge models, the inference mechanism can learn after diagnosis. A new case about the diagnosis of the object is added to the case-base so that similar faults can be diagnosed in future. When enough cases are assembled, some similar cases can be used to generate new rules and construct new neural models.

The following are the approaches of a diagnosis based on each knowledge model:

(a) Computation of predicated state features based on the operate model.

When the observed behavior of an object to be diagnosed is inputted the predicated behavior of the object can be derived by the operate model using predefined algorithms: fuzzification, MIN-MAX fuzzy inference according to the fuzzy rules learned by GA, defuzzification. The results of fuzzification are memberships [0,1] of fuzzy predicates of input variables. $Z_{mn}=\mu_{A_{m_n}}(I_m)$, A_{mn} is the nth category of fuzziness of I_m. The results of Min-Max fuzzy-inference are AND-operation of premises of fuzzy rules (minimum of memberships of input variables), and OR-operation of conclusions of fuzzy rules (maximum of conclusions derived by different fuzzy rules). The results of defuzzification are

$$O_k=\sum_1^n Y_{kl}\times C_{kl}\Big/\sum_1^n Y_{kl},$$

C_{kl} is the middle value of the triangular membership function of A_{kl}, A_{kl} is the lth category of fuzziness of O_k.

(b) **Rule-based diagnosis**

For an object to be diagnosed, abnormal symptoms are determined according to the difference between observed state features and predicated ones derived by the operate model. Then a diagnostic rule is searched in the rule-base, whose premises match the abnormal symptoms of the object, the conclusion of the rule is derived as the result of the diagnosis.

(c) **Neural network-based diagnosis**

For an object to be diagnosed, according to its abnormal features, a neural model is selected from the knowledge base. The values of the features which are normalized into [0,1] are inputted into the nodes of the input layer of the multiple-layer preceptron network; the outputs of the output layer of the network are calculated as the results of the diagnosis.

(d) **Case-based diagnosis**

For an object to be diagnosed, the abnormal signs of its state features are determined according to the difference between observed features and predicated ones derived by the operate model. The binary string of the signs of features is used as the index for case retrieve in the case-base. Though there may be 2^m classifications in the case-base at most, a case classification can be searched in the binary trees by m times of bit matching. If there is only one case in the case classification, then the diagnosis of the case is adopted as the result of the diagnosis of the object. If there is more than one case in the case classification, the distances between the object and each case in the case classification are calculated in form of

$$D_i=\frac{1}{n}\sqrt{\sum_{j=1}^n (f_{ij}-f_{oj})^2}\quad \mathrm{i}=1,\cdots,N$$

where f_{ij} – the jth feature of the ith case in the case classification, f_{oj} – the jth feature of the object, n – the number of features of a case, N – the number of cases in the case classification. The nearest case is selected; its diagnosis is adopted as the result of the diagnosis of the object.

5. CONCLUSIONS

Fault diagnosis is an important area of research in *AI*, and a key technique in Computer-Integrated Manufacturing Systems. The bottleneck in building an expert system for fault diagnosis is knowledge acquisition and multiple knowledge modeling. Traditional expert system shells cannot deal with automatic modeling of fault diagnosis and diagnosis based on multiple knowledge models. In this paper, a new expert system shell for fault diagnosis is discussed. The new expert system shell can learn multiple knowledge models (object model, rules, neural network, case-base) using genetic algorithms, rough sets, multi-layer perceptron networks and case-learning. It can diagnose hierarchically (from general to specific) based on multiple knowledge models. The expert system shell has been used in a steel-making company in Shanghai. It has been verified to be more effective than traditional expert system shells for fault diagnosis.

References

1. J. de Kleer and B.C. Williams, "Diagnosis with behavioral models", *IJCAI-89*, (1989) pp. 1324–1330.
2. Jie Yang, *Prioritized Justification-based Nonmonotonic Logic and its Applications* (Shaker Verlag, Aachen, Germany, 1994).
3. M.P. Feret and J.I. Glasgow, "Combining Case-Based and Model-Based Reasoning for the diagnosis of Complex

https://doi.org/10.1017/S0263574701003460 Published online by Cambridge University Press

Devices", *Journal of Applied Intelligence* **7**, No. 1, 57–78 (1997).
4. A.R. Golding and P.S. Rosenbloom, "Improving accuracy by combining rule-based and case-based reasoning", *Artificial Intelligence* **87**, No. 1-2, 215–254 (1996).
5. N. Shan et al., "GRG : Knowledge Discovery Using Information Generalization, Information Reduction, and Rule Generation", *7th International Conference on Tools with Artificial Intelligence*, Washington, D.C., USA (1995), pp. 372–379.
6. C.T. Lin et al., "Fuzzy adaptive learning control network with on-line neural learning", *Fuzzy Sets and Systems* **71**, No. 1, 25–45 (1995).
7. M. Mohammadian et al., "A Case Study of Knowledge Acquisition: from Connectionist Learning to an Optimized Fuzzy Knowledge Base", *IEEE 2nd International Workshop on Emerging Technologies and Factory Automation* (1993) pp. 106–111.
8. M.H. Lim et al., "A GA paradigm for learning fuzzy rules", *Fuzzy Sets and Systems* **82**, 177–186 (1996).
9. Z. Pawlak, "Rough Set", *International Journal of Computer and Information Science* **11**, No. 5, 341–356 (1982).
10. Z. Pawlak, *Rough Set, Theoretical Aspects of Reasoning About Data* (Dordrecht, The Netherlands: Kluwer, 1991).
11. A. Mrozek, "Rough Sets and Dependency Analysis among Attributes in Computer Implementations of Expert's inference models", *International Journal of Man-Machine Studies* **30**, 457–473 (1989).
12. J.H. Chen, "Algorithm of Knowledge Discovery and Its Applications", *Ph.D. Dissertation* (Shanghai Jiao-tong University, China, 1998).
13. Ron Sun, "Commonsense reasoning with rules, and connectionist models: A paradigmatic comparison", *Fuzzy Set and Systems*, **82**, 187–200 (1996).
14. L.R. Wedsker, *Hybrid Neural Network and Expert Systems* (Kluwer Academic Publishers, 1994).

https://doi.org/10.1017/S0263574701003460 Published online by Cambridge University Press