Rough Sets and Genetic Algorithm Using for Diagnostic Problems Solving

Sakena Saied Alsadig ALTAHR, Štefánia GALLOVÁ, *Member, IAENG*

Abstract—**This paper provides an effective approach of rough sets principles and genetic algorithm applying for real-world diagnostic problems solving of machine tools equipments. The basic source of information is vibration signal derived from sensor that is placed on rotating part point of machinery equipment. Rough set theory principles is an effective tool for indiscernibility expression of judged states Genetic algorithm makes possible to provide an appropriate solution with missing values presence and without existing of exhaustive knowledge about judged situation. Our designed approach effectively connects and combines both, rough set and genetic algorithm principles. Experimental results confirm well-found of approach in comparing with classical generally used approaches.**

Index Terms— **genetic algorithm, rough set, chromosome, training set, classification.**

I. INTRODUCTION

There are some well-known strategies how to deal with real-world diagnostic problems of machinery equipments within optimal technological operation satisfying in presence first of all incomplete or insufficient value features within realized classification problem. Rough sets theory principles provide an effective tools how to deal with this type of problems of practice. Genetic algorithm makes an opportunity to provide an appropriate solution with incomplete and a priori not available part of information about judged state of equipment. We have applied some generally known investigated approaches to this task. Next, we use an effective and reliable improved approach. Classification process with missing, incomplete and nonsufficient knowledge about solved situation is the problem of real-world tasks.

This method provides the best results within realized experiments. This approach is more sophisticated and robust than other ones. Analysis and implementation of this approach provides the best results from another used standard methods. [1],[3],[5],[7].

Sakena Saied Alsadig Altahr is PhD student, Faculty of Science, Institute of Computer Science, Pavol Jozef Šafárik University in Košice, 040 00 Slovak Republic,

S. Gallová, Faculty of Science, Institute of Computer Science, Pavol Jozef Šafárik University in Košice, 040 00 Slovak Republic, (e-mail: stefania.gallova@upjs.sk)

II. PROBLEM DESCRIPTION

A. Fault diagnosis using rough set theory

We analyze nine relevant states that content attributes derived from the vibration signals and that provide the characteristics of signal within chosen sampling points carefully defined to collect vibration signals. The information table provides a convenient tool for description of objects in terms of their attribute values.

We have an information $S = \{U, A, V\}$, where U is a finite set of objects:

$$U = \{x_1, x_2, \dots ,x_n\}, \tag{1}$$

and A is a finite set of attributes, i.e. *a: $U \rightarrow V_a$* for $a \, \epsilon \, A$, where Y_a is the domain of a, and $V = U_{(a \epsilon A)}V_a$. A decision table is any information table S of the form $S = (U, C \cup \{d\}, V)$, where d *is not ϵ C* is a distinguished attribute called decision attribute. The elements of C are called condition attributes.

Let $S = (U, A, V)$ be an information table, every non-empty subset of attributes $B \leq A$ is associated an indiscernibility relation on $U,$ denoted by *IND(B):*

$$IND(B) = \{(x,y) \, \epsilon \, U^2 : \text{ for each } a \, \epsilon \, B, \; a(x) = a(y)\} \tag{2}$$

They are *B*-indiscernible.

B. The Reason for Genetic Algorithm Using

Comparable solutions are very slow and often very complicated. Genetic algorithms provide a hybridization with another solutions that are exist and they are rich from implementation point of view due to the large and mushroom domains number. Surprisingly, they provide in some specific domains more appropriate solutions. Genetic algorithms effectively handle with uncertainty and high-dimensionality.

III. PROBLEM ANALYSIS

We have: Normal state - sample point denoted 1;
(x-coordinate axis: time [m/s]; y-coordinate axis-amplitude)
(x-axis: Frequency [Hz]; y-coordinate axis-FFT spectrum)
[1], [4]

$$BN_B(X) = \text{upper } B(X) - \text{lower } B(X) \tag{3}$$

$$\text{lower } g_B(X) = \frac{|BX|}{|U|} \qquad \& \qquad \text{upper } g_B(X) = \frac{|\overline{B}X|}{|U|}$$

(4)

A. Information content of coefficients

From detailed and large analysis of derived vibration signal from semsor that is placed on chosen rotating point of machine tool. We can summarize, that the coefficient Δz_{01} is the coeficient of the change of the whole signal level and means the following [2], [3]

$\Delta z_{01} > 0$.....arising of signal level
$\Delta z_{01} = 0$.....level is without change
$\Delta z_{01} < 0$.....decreasing of signal level

The coefficients Δz_{02} reflects the change of the spectrum shape. We have:

$\Delta z_{02} > 0$.....high-frequency part of spectrum enlarges
$\Delta z_{02} = 0$.....spectrum shape is without changes
$\Delta z_{02} < 0$.....low-frequency part of the spectrum enlarges

$\Delta z_{03} > 0$.....breadth enlarging of zone
$\Delta z_{03} = 0$.....breadth of zone is without changes
$\Delta z_{03} < 0$..... breadth reducing of zone

B. Detectionn and classification of deviation

The germ, source of the breakdown can arises in this case, when the coefficients values start to change. The deviation detection is the best to realize on the basis of finding out that the coefficient of the level of diagnostic signal Δz_{01} oversteps chosen limit value d, that means:

$$|\Delta z_{01}| > d \qquad (5)$$

C. Learning process

After deviation finding out, learning process automatically starts for the reason to achieve new reference values. They should be achieved only in this case, when, after deviation appearing, signal stationarity will redress on the new level. For example, deviation will arise by including a new gear into gear-unit, and after again appearing the new revolutions, signal of monitoring sensor is again stationary.

To determine a stationarity of signal, the most appropriate method is to test a trend of measured values of diagnostic coefficients. If the result is corresponding , that means, the trend of signal is zero, we can compute new reference values from measured file of values. Learning process starts together with monitoring starting, because in this time we don't know values of reference coefficients. External memory for long-time saving of location of breakdown history.

Genetic algorithms belong to the field of evolutionary computation. They model a natural evolution by tools of genotype. Every candidate solution can be expressed as a chromosome. One chromosome holds a set of genes. One gene is for every variable to be optimized. We introduce a fitness function which evaluates how well every chromosome matches the ideal solution. We have three operators such as selection, crossover and mutation that are applied within evolution process.

The selection operator uses the fitness values to pick a set of chromosomes to survive through to the next generation. Crossover operator creates offspring chromosomes by combining genetic material from a set of parents. Mutation operator randomly alters the genes of a single chromosome to introduce novel genetic material.

procedure GA
begin
    t=0;
    initialize P(t);
    evaluate structures in P(t);
    while termination condition not satisfied do
    begin
        t=t+1;
        select P(t) from P(t-1);
        mutate structures in P(t);
        crossover structures in P(t);
        evaluate structures in P(t);
    end
end

Fig 1. Classical genetic algorithm.

IV. PROBLEM SOLVING

A. Data analysis rough sets

Chromosome i.e string, contents components group of optimalized object which influences their attributes. Chromosome string includes its information content. The goal is to find optimal values of string components, as example we have:
S={4,9,1,2,7,5,6}
  - string components values
S={f_1, f_2,...,f_n}
  - number of elementary frequencies
S={v_1, v_2, ...,v_n}
  -values of judged parameters
S={p_1, p_2, ...,p_n}
  - order of passed points (places)

Population contents a set of strings. The number of solved strings is from 30 to 90 ones. The objective function measures the successfulity of each string, each potential solution. Fitness function is minimized or maximized, i.e. we judge the „fitness". Objective function is defined by user and represents the core of the problem which is optimized. Operation implemented within genetic algorithm implementation are the following:
- *selection*

The aim is to choose a sufficient number of strings (parents) that are modified by genetic operators and to choose another ones that will go without changes to the new generation. There are many types of selection process (randomly realized selection, selection by successfulness, tournament or roulette selection, and another ones. The rule is that more succesfull strings have a higher chance to be a

member of next generation. Randomly chosen selection operation with applied some a priori information (if exist) provides good results [2], [3].

- *crossover*

In the whole population or in only within its part. The type of machine tool and the place for sensor measurement decide about type of crossover operation.

- *mutation*

Is realized within population with probabilities in the range of 0.01 - 10%. In solved tasks, the mutation parameter was experimentally delicately setted.
Generally, its values is about 0.55.

B. Problem solving in practice by genetic algorithm.

Model of optimized object first of all contains objective function accounting, program, simulation and so on. Secondly, it contains a model of evolution (genetic algorithm) and as a result of this process we have an appropriate solution.

The aim of our effort is to evaluate and compare the efficiency of some known effective algorithms for missing value handling implementation with our one suggested approach to missing value handling algorithm implementation. Chromosome representation is first basic step. There are several possible chromosome representations, such as binary and non-binary representation. Non-binary representation is real, integer, tree, permutation and another ones possibly complex data structures.

We solve generation size and initialization of the generation. A solution is encoded as the following data structure using binary code representation. Fitness function provides numerical expression of how fit the solutions that is encoded in a particular chromosome is. We can realize complete or approximate function evaluation. Fitness function is the rating chromosome in terms of their ability to solve our problem. There is an illustrative example of chromosome representation:

Chromosome X_i: 10010001 with genes: 1,0,0,1,0,0,0,1.
                  10110000
                  10110010

Genetic algorithm:
   generate initial population G(0);
   evaluate G(0);
   for (t=1; solution is not found, t++)
      generate G(t) using G(t-1);
      evaluate G(t);

Fig. 2. Part of Genetic algorithm implementation.

Original chromosome	gene	gene	gene	gene	gene	gene	gene	gene
Mutation 1	gene	gene	gene	gene	gene	gene	gene	gene
Mutation 2	gene	gene	gene	gene	gene	gene	gene	gene
Mutation 3	gene	gene	gene	gene	gene	gene	gene	gene

Fig. 3. Chromosome – mutation operation.

Parent 1	gene	gene	gene	gene	gene	gene	gene	gene
Parent 2	gene	gene	gene	gene	gene	gene	gene	gene
Offspring 1	gene	gene	gene	gene	gene	gene	gene	gene
Offspring 2	gene	gene	gene	gene	gene	gene	gene	gene

Fig. 4. Chromosome – crossover operation.

TABLE I
CHROMOSOME – LOWER AND UPPER APPROXIMATION OF GENE

Valid genes	lower A(X4)	Lower A(X3)	lower A(X2)	lower A(X1)	upper A(X4)	upper A(X3)	upper A(X2)	upper A(X1)
gene1	0	0	0	0	0	0	1	1
gene2	0	0	0	0	0	1	0	1
gene3	0	0	0	0	1	1	0	0
gene4	0	0	0	1	0	1	1	1
gene5	0	0	0	1	1	1	0	1
gene6	0	0	1	0	0	0	0	1
gene7	0	0	1	0	1	0	1	0
gene8	0	0	1	1	1	0	1	1
gene9	0	1	0	0	0	0	0	0
gene10	0	1	0	0	0	0	1	0
gene11	0	1	0	0	0	1	0	1
gene12	0	1	1	0	0	1	1	1
gene13	1	0	0	1	1	0	0	1
gene14	1	0	1	0	1	0	1	0

$$\Delta = \sum_{i=1}^{m} \sum_{u_h, u_k \in X_i} DISTANCE(u_h, u_k) \quad (6)$$

Selection process is implemented by fitness function on the basis of proportionality copies the individuals in the current generation. That means, "better solution" go to the next generation and the „poorer solutions" are canceled:

$$s(x_i) = \frac{f(x_i)}{\sum_{i=1}^{n} f(x_i)} ; \quad i=1,...,n \text{ (generation)} \quad (7)$$

V. EXPERIMENTS

Rule confidence definition is following [2], [3], [7].
        confidence (X→Y) = support (X→Y) /support(X)

A. Support and confidence measures

The task is to find all possible rules $X \wedge Y \Rightarrow Z$. They have minimum support. Support is a probability that realized transaction which contains {X,Y,Z}. Confidence is a conditional probability where a transaction with {X,Y}, also contains Z.

TABLE II
MEASURES

Change	Components	Component frequency	Support
101	A, D	{B}	50%
100	A, B, D	{A}	75%
111	A, C	{D}	50%
001	B, E, F, H	{A,D}	50%

We have the form of rule as follows:

"$X \rightarrow Y$"

" Amplitude $\rightarrow$ Range (support, confidence)"

If we have the case with a minimum support equal 50%, and minimum confidence equal 50%, we can express the following:

$A \Rightarrow D$ rule (50%, 66,66 %)

support = support ({A,D}) = 50%

confidence = support ({A,D})/support ({A}) = 66,66%

$D \Rightarrow A$ (50%, 100%)

B. Used procedure

The rule is that any subset of frequent components must be also frequent. If {A,D} is a frequent component, then also {A} should be a frequent component and also {D} should be a frequent one. That means, each subset of frequent component must be also a frequent component. We search for frequent component by iteration process with cardinality from 1 to k amount of components. Then we apply a frequent component for generation of association rules. Frequent one cannot belongs to the subset that contains the frequent k-component set.

We have:

L_{k-1} self-joining

insert into C_k

select $p.component_1,\ p.component_2,..., p.component_{k-1},$
$q.component_{k-1}$

from $L_{k-1}p,\ L_{k-1}q$

where $p.component_1 = q.component_1,..., p.component_{k-2} = q.component_{k-2},$
$p.component_{k-1} < q.component_{k-1},$

We can realize reducts with minimal rules (or cardinalities) and minimal pairwise intersections. [2], [3]

TABLE III
FIRST GENERATION

Chromosome $X(i)$	Fitness function $f(X_i)$
10001000	136
00011000	24
10001110	142
00001001	9
00011010	26
10001001	137
10001011	139
00001000	8
10101011	171
Total =	792
Average =	88

We have the best individual 10101011 with its fitness value equal 171. In the first generation, we have a total sum of fitness values of individuals equal 792 and average fitness value is equal 88. The individual „10101011" will be selected for the next generation with the probability witch is the highest in comparing with selection probabilities of another individuals in solved generation. The probability of the most succesfull individual has the value equal to the ratio 171/792 (fitness value of the best individual/total sum of individuals fitness values for one generation).

Problem solving example by proposed genetic algorithm is following. We have nine items, such as signal level, high-frequency spectrum part, low-frequency, frequency range,...We chose the highest fitness value of the next item. Chromosome with eight bits string where each bit corresponding to the item. If the corresponding item is included, elementary bit is equal 1. If corresponding item is not included, bit is equal 0.

"Fit chromosome =

= quality of elementary frequency cycle parameters"

VI. CONCLUSION

Rough set theory principles is an effective tool for indiscernibility expression of judged states.

Model implementation was applied on real-world collaborative sets of data. It works better than comparable models. The present analysis and implementation of this approach provides the best results from another used standard methods. The importance of genetic algorithms using is in the following cases. In real-world situations we often have to solve the hard optimization problems that are usually not possible to solve by classical approaches. In practice, we usually have:

1) nonlinear, multidimensional, nonconvex, multimodal functions;
2) combinatorial tasks solving;
3) non-analytical problems;
4) tasks with many different constraints;
5) multicriterial tasks;

Solved diagnostic problem is large, complex and usually contents several previous attributes.

REFERENCES

[1] B. Caruanna, A. Niculescu-Mizil: "Data mining on Metric Space, "An Empirical Analysis of DD, pp.319-352, (2013).

[2] A.A.A. Salem: "Artificial intelligence principles usng for real-world diagnostic problems solving," PhD.Thesis, University of P.J. Safarik Kosice, Inst.of Computer Science , Slovak Republic, Kosice, 2013.

[3] S.Gallova.: "Contribution to Icomplete and Noisy Information Problem Solving by Artificial Intelligence Principles Applying. Lecture Notes in Engineering and Comp. Science, Nevswood Limited, London, 2010.

[4] J.W. Grzymala-Busse.:"On the unknown attribute values in learning from examples, Proceedings of the ISMIS-91, 6th International Symp. on Methodologies for Intelligent Systems, Charlotte, North Carolina, Lecture Notes in Artificial intelligence, volume 542, Springer-Verlag, Berlin, Heidelberg, New York, pp.368-377, (1991).

[5] J. Knonenko, and B.Roskar.: "Experiments in automatic learning of medical diagnostic rules," Technical Report, J.S.Institute, Ljubljana. (1984)

[6] B. Marlin: "Modeling user rating profiles for collaborative filtering", Proceedings of the 17th Annual conference on Neural Information Processing Systems (NIPS-2003), (2003).

[7] B. Marlin: "Collaborative filtering," A machiine learning perspective. University of Toronto, 2004.

[8] J.R. Quinlan.: "C4,5:Programs for Machine Learning," Morgan Kauf. Publshers, San Matteo, CA, (1993).

[9] R. Nicole: "Title of paper with only first word capitalized," J. Name Stand Abbrev. (in press).