http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 6 (6): 865-871, 2007
ISSN 1812-5638

The Method of the Least Reduction in Oil Reservoir Based on Rough Set Particle Swarm

[1,3]Hongjie Liu, [1]Boqin Feng and [2]Jianjie Wei
[1]Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an 710049, China
[2]College of Mathematics and System Science, Xingjiang University, Urumqi 830046, China
[3]Geophysical Research Institute Xinjiang Oil-Field, Urumqi 830013, China

Abstract: In the oil reservoir prediction, it is not that the more index variables of seismic data, the better effect of classification is. On the contrary, the classification accuracy will reduce because of the redundant index variable influenced by the calculation error. Therefor, a method of the least reduction of oil reservoir data is presented in this study. In this method, we directly adopt the dependency of rough set and PSO algorithm by binary encoding, which make the two algorithm organic affiliated. The actual application on the data of oil reservoir not only shows that using this method presented in this paper achieves very notable effect, but also shows that this method has great actual meaning to further construct higher efficient attribute reduction method.

Key words: Rough Set, PSO, attribute reduction, oil reservoir prediction, dependency, fitness

INTRODUCTION

The classical article Rough Sets (Pawlak, 1982) has been published by Z.Pawlak. Thereafter, the monograph (Rough Sets-Theoretical Aspects of Reasoning about Data) has also came out by Z.Pawlak, which indicates that research on rough set theory and its applications come into hotspot era. The first international academic conference on rough set theory was held in Poland in 1992, which concluded the achievement of theory and practice in this period. Rough set theory has been extensively applied in many fields such as information systems analysis, artificial intelligence, pattern recognition, oil reservoir prediction and so on. (Pawlak, 1999; Liu *et al*., 2006; Zhang and Qiu, 2007; Liu, 2003; Li and Li, 2003).

At present, attribute reduction is a key research in rough set, which studied by many experts in the world. Though its theory research has made great progress, the specific implement method on attribute reduction is little. The methods at present have achieved effect on some problems, but up to now, there is not a acknowledged, effective attribute reduction method. Finding out a least reduction of decision table is a NP-hard problem (Wong and Ziarko, 1985) which has been proved by Wong and Ziarko (1985) however, the combinatorial explosion problem of attribute is the main cause of NP-hard.

Particle Swarm Optimization algorithm (PSO) is a new evolutionary computation algorithm presented by Kennedy and Eberhart (1995) which reappears swarm intelligence. This algorithm adopts velocity-position searching model and its main advantages are that PSO is easy to implement and there are few parameters to adjust. It not only holds the swarm intelligence background of traditional evolutionary computation, but also has many favorable optimal performance.

In the oil reservoir prediction, it is not that the more index variables of seismic data, the better effect of classification is. On the contrary, the classification accuracy will reduce because of the redundant index variable influenced by the calculation error. Therefor, a method of the least reduction of oil reservoir data is presented in this study. In this method, we directly adopt the dependency of rough set and PSO algorithm by binary encoding (Kennedy and Eberhart, 1997), which make the two algorithm organic contact. Its advantages are as follows: firstly, this algorithm is more direct as it is no need to determine core by the discernibility matrix. Secondly, the decision table can be inconsistent, however, it must be consistence by the algorithm using discernibility matrix in other papers, or it could not reduce attribute. The experiment shows that the application effect is very notable, it not only achieves anticipation, but also has high fitting precision.

PROBLEM DESCRIPTION

Basic notions of rough set (Liu, 2003; Li and Li, 2003): Knowledge and knowledge base: Suppose U is a finite set of the interested objects, named universe, a group

Corresponding Author: Dr. Hongjie Liu, Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an 710049, China

allocation of U is called a knowledge base about U. Suppose R is a group of equivalence relation of U, then knowledge base can notes as K=(U,R).

If $P \subseteq R$ and $P \neq \phi$, then $\cap P$(the intersection of all equivalence relation in P) is a equivalence relation, which is indiscernibility relation in P namely ind(P). Therefore, U/ind(P) is the knowledge related with the equivalence relation group P, called the basic knowledge P about U in K.

The lower approximation of set and positive region: Given knowledge base K = (U,R), for each subset $X \subseteq U$ and a equivalence relation $R \in ind(K)$, the upper and lower approximation of set X about R are respectively defined as follows:

$$\underline{RX} = \cup \{Y \in U/R | Y \subseteq X\} \quad \text{and}$$
$$\overline{RX} = \cup \{Y \in U/R | Y \cap X \neq \phi\}$$

$Pos_R(X) = \underline{RX}$ is called positive region of set X about R.

Information system, decision table and consistent decision table: Information system is defined as a quad S = (U, A, V, f), where U is a finite set of objects. $U = \{u_1, u_2, \ldots, u_n\}$ is named universe. A is a finite set of attribute,

$$A = \{a_1, a_2, \ldots, a_m\};\ V = \bigcup_{a \in A} V_a$$

V_a is the range of attribute a f: $U \times A \rightarrow V$ is a information function, which endows a value to each attribute of each subject, i.e., $\forall a \in A$, $x \in U$, $f(x,a) \in V_a$. Usually use S = (U,A) to instead S = (U, A, V, f).

Suppose S = (U,A) is a information system and $C, D \subseteq A$, $C \cap D = \phi$,C and D are, respectively the condition attribute set and the decision attribute set of A, so S is shown as T = (U, A, C, D), which is a decision table.

Function d_x: $A \rightarrow V$, let $d_x(a) = a(x)$, where $a \in A$, $X \subseteq U$, $x \in U$, dx is called a decision rule of T. If $a \in C \subset A$, then dx|C is noted as the condition part of decision rule; if $a \in D \subset A$, then $d_x|D$ is noted as the conclusion part of decision rule. For every individual, if $y \neq x$, $d_x|C = d_y|C \rightarrow d^x D = d_y$, then d_x is consistent, or not.

Consistent decision rule indicates that the same condition value must implicate the same decision value, i.e., the decision rule entirely depends on the condition value.

Dependency: Suppose T = (U, A, C, D) is a decision table, D depends on C by degree $k(0 \leq k \leq 1)$ in T, noted as $C \rightarrow_{T,K} D$, where K = card (POSc (D))/card (U), card(S) is the base number of S.

Basic notion of PSO algorithm by binary encoding: (Kennedy and Eberhart (1995); Kennedy and Eberhart, 1997): Suppose that there are m particles in D-dimensional search space. The position of the ith particle is presented as

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{iD}), \qquad I = 1, 2, \ldots, m,$$

which is potential solution, input it into optimal object function, we can get the corresponding fitness value, then evaluate the quality of x_i by it; the best position experienced by the ith particle is named the individual best position and presented as

$$p_i = (p_{i1}, p_{i2}, \ldots, p_{iD})$$

at the same time, each particle has its own velocity

$$V_i = (v_{i1}, v_{i2}, \ldots, v_{iD}),$$

the best position experienced by all particles is named the global best position and presented as

$$P_{gi} = (P_{gi1}, P_{gi2}, \ldots, P_{giD})$$

this corresponding fitness value F_g is the global best position in history. For each generation particle, its dth dimension $(1 \leq d \leq D)$ updates by following equations.

$$V_{i,j}(t+1) = \chi(v_{i,j}(t) + c_1 r_{1,j}(t)(P_{i,j}(t) - x_{i,j}(t)) + c_2 r_{2,j}(t)(P_{gj}(t) - x_{i,j}(t))) \tag{1a}$$

$$x_{i,j}(t+1) = \begin{cases} 0, r_3, \geq Sig(v_{i,j}(t+1)) \\ 1, \quad else \end{cases} \tag{1b}$$

Where, χ is convergence constant which effectively improves the convergence rate c_1 and c_2 called acceleration constant are two positive constant r_1 and r_2 are two random values in the range [0,1] Sig(x), is a fuzzy function defined as

$$Sig_x = \frac{1}{1 + \exp(-x)}$$

During the iteration, $|v_{ik}| \leq v_{max}$, where $vik(1 \leq k \leq D)$ is the component of particle velocity $V_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$, if $v_{ik} > v_{max}$, then $v_{ik} = v_{max}$, if $v_{ik} < -v_{max}$, then $v_{ik} = -v_{max}$.

Least attribute reduction based on rough set particle swarm

Principle description: The position of the particle adopts relative visual and conventional binary encoding, obtaining the least attribute reduction is to search the least attribute subset from N condition attributes. The encoding project is that using a binary string length of N to present an individual, each bit corresponds a condition attribute, 1 represents the chosen subset containing the corresponding attribute, 0 represents not.

Definition 1: Suppose there are four condition attributes $\{a_1, a_2, a_3, a_4\}$, a reduction may be $\{a_3, a_4\}$,then X_{pb}= 0011, where X_{pb} is the binary encoding of this particle in position X.

Definition 2: If R is the subset of condition attribute, D is the decision attribute, then the dependency of D for R is as follows:

$$k = card(POS_R(D))/card(U)$$

Definition 3: Suppose R is a condition attribute subset corresponding with rough particle position *x* (binary encoding), then the fitness of rough particle is as follows:

$$f(x) = card(POS_R(D))/card(U)$$

Definition 4: The least reduction of rough particles in consistent decision table is evaluated by that: the fitness value of rough particle is 1 and the attribute is the least in R, i.e., the number of 1 is the least in X_{pb} in definition 1.

Definition 5: The least reduction of rough particles in inconsistent decision table is evaluated by that: the fitness value of rough particle is the maximum and the attribute is less in R (i.e., the least reduction could not guarantee the number of attribute in R is the least at one time), that is the number of 1 is less in X_{pb} in definition 1.

Theorem 1: In a consistent decision table, suppose R is a condition attribute subset corresponding with rough particle position *x* (binary encoding), if the fitness value of rough particle is 1 and the number of 1 is the least in X_{pb}, then R is the least reduction.

Prove: (omit).

Theorem 2: In a inconsistent decision table, suppose *R* is a condition attribute subset corresponding with rough particle position *x* (binary encoding), if the fitness value of rough particle is the maximum and the number of 1 is less in X_{pb}, then R is the least reduction.

Prove: (omit).

Algorithm:

Rough set particle swarm algorithm:

- Initialize the decision table, extract the number of samples and condition attribute from the inputs and calculate the equivalence class uindc of each condition attribute and the equivalence class uindd of decision attribute.
- Initialize the particle swarm (population size is m), including the settings of random position and velocity.
- Evaluate the particle swarm.

Evaluate the fitness value of each particle.

For each particle I, compare current fitness value with the fitness value of the best position in the $_{history\ Pi}$, If the current fitness value is better, then set current fitness value as the new P_i.

For each particle, compare the fitness value of all particles in the swarm with the global best position in the history P_g, if its fitness value is better, then set the current position as the new P_g.

- Update the velocity and position of particles according to equation (1).
- If it does not reach the termination condition, which is the enough good fitness value or a given maximum iteration Gmax, then return to (3), or go to (6).
- The least reduction is the condition attribute set of the global best position P_g.
- End.

Simulation experiment and example analysis

Simulation experiment: To verify the feasibility of the least attribute reduction algorithm based on rough set and PSO in this paper, choose the decision table given by Table 1 (Liu, 2003) as simulation experiment. The program processes are shown in the Fig. 1 and 2.

Table 1: Results of decision table

U	a	b	c	d	e
u_1	1	0	2	1	0
u_2	0	0	1	2	1
u_3	2	0	2	1	0
u_4	0	0	2	2	2
u_5	1	1	2	1	0

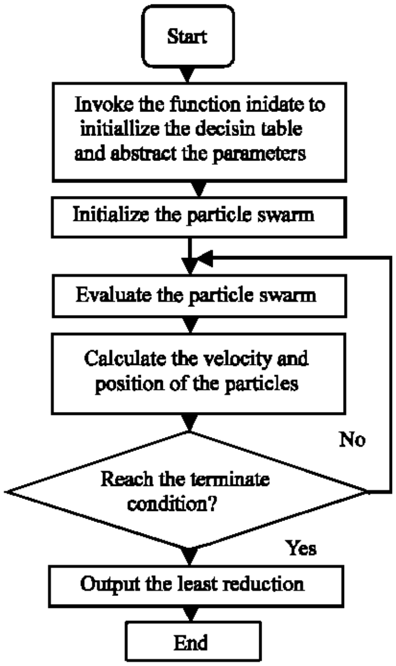

Fig. 1: The flow chart of the least attribute reduction of rough set particle swarm

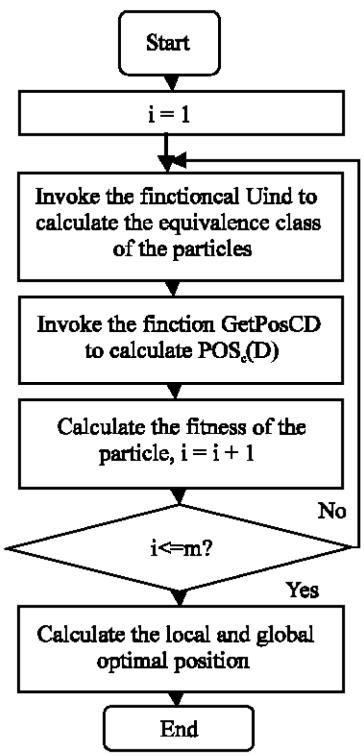

Fig 2: The flow chart of the evaluation of rough set particle swarm

To calculate the fitness of the rough set particles, design and operate the main functions as follows:

Function [sampleNum poslen uindc uindd] = inidata
Function description: choose the number of samples and condition attribute from the inputs and calculate the equivalence class uindc of each condition attribute and the equivalence class uindpd of decision attribute by invoking function calUindC, calculate the equivalence class uindd of decision condition by invoking function calUind.

Function Uind=calUindC(table,valueNum)
Function description: calculate the equivalence class division of all attributes in attribute table.

Input parameter: condition attribute or decision attribute in attribute table, the number of class named valueNum in each row of attribute table.

Function uind = interPart(ind1,ind2)
Function description: calculate the equivalence class ind1 and refinement equivalence class of ind2.

Function uind = calUind(uindc,ppos)
Function description: calculate the equivalence class division of attribute set ppos by invoking function interPart.

Input parameter: the equivalence class array uindc of each attribute in attribute set A, the given attribute set ppos (the encoding of particle position).

Function posCD = GetPosCD(uindc,uindd)
Function description: calculate $POS_c(D)$

Input parameter: the equivalence class uindc of condition attribute and the equivalence class uindd of decision attribute.

Convergence constant χ = 0.7298, $C_1 = C_2$ = 2.0, the maximum velocity of particles v_{max}= 6.0, the number of particles popsize=8, the maximum iterative number maxgap =50.

The experiment result shows that the average number of attribute converges to 2 in the optimal solution, where the position encoding of attributes are 1010 and 0011. This result consists with the reduction {a,c} and {c,d} in reference (Liu, 2003).

Example analysis: The earthquake data in XX area is shown in Table 2, where condition attributes are

Table 2: The samples of oil reservoir

No.	1	.	30	31	.	47
c_1	6.5628	.	3.7115	10.3489	.	4.468
c_2	20.6578	.	29.8993	28.4919	.	65.4253
c_3	58.6703	.	80.8227	23.7523	.	-19.252
c_4	106.8865	.	18.904	108.2298	.	19.1269
c_5	64.7361	.	76.9923	54.8152	.	0
c_6	-89.8571	.	-11	-83.5714	.	-7.8571
c_7	85.2784	.	15.7236	89.9723	.	19.8422
c_8	10725.47	.	582.7688	12692.06	.	666.104
c_9	295.1573	.	-207.4411	27.8702	.	5429.87
c_{10}	-629	.	-77	-585	.	-55
c_{11}	60227	.	1559	53013	.	2149
D	1	.	2	1	.	2

Arc-length (c_1),
Arc-Inst-Freq(c_2),
Avg-Inst-Phase(c_3),
Abg-RefI-Str(c_4),
Energy-Half-Time(c_5),
Mean-amplitude(c_6),
RMS-Amplirude(c_7),
Slope-Half-Time(c_8),
Slope-Inst-Freq(c_9),
Total-Amplitude(c_{10}),
Total-Energy(c_{11})
i.e .,

$$C = \{c_1, c_2, {}_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}\}$$

let decision attribute $D = \{d\}$, $d = \{d_I = I, I = 0,1\}$, where 0 and 1, respectively represent oil layer and dry layer.

Process the data of Table 2 by the method mentioned in this paper after discretization. Choose No. 1 to 30 as training samples, because there are 11 condition attributes, we select the number of particle swarm popsize = 20; convergence constant, $\chi = 1$; $C_1 = C_2 = 2.0$; the maximum velocity $V_{max} = 6.0$.

The experiment result shows that the number of average attribute in optimal solution converges to 4, where the position encoding of attributes are 01001001100 and 01001010100, so the reduction are $\{c_2, c_5, c_8, c_9\}$and $\{c_2, c_5, c_7, c_9\}$.

Comparison with genetic algorithm: Respectively use the rough set particle swarm algorithm and the genetic algorithm to process the data of Table 2.

The running results of rough set particle swarm algorithm are shown in Fig. 3 and 4, convergence constant $\chi = 1$; $c_1 = c_2 = 2.0$;the maximum velocity $v_{max} = 6.0$.

The running results of genetic algorithm are shown in Fig. 5 and 6 where PC = 0.5; Pm = 0.1; adopt optimal preserved strategy; selection operator adopts single-point crossover; mutation operator adopts single-point mutation.

From the Fig. 5 and 6 we can see that the maximum dependency in the two algorithms don't reach 1, the

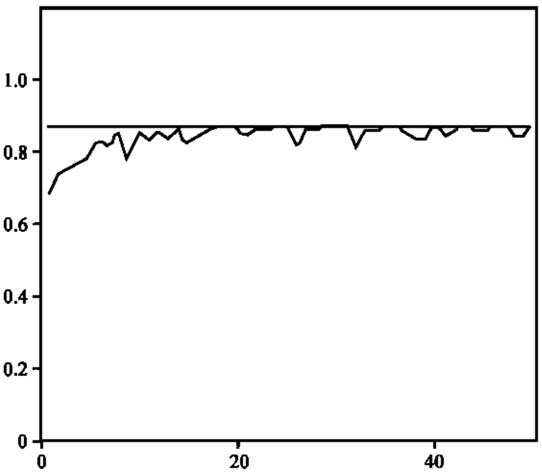

Fig 1: The flow chart of the least attribute reduction of rough set particle swarm

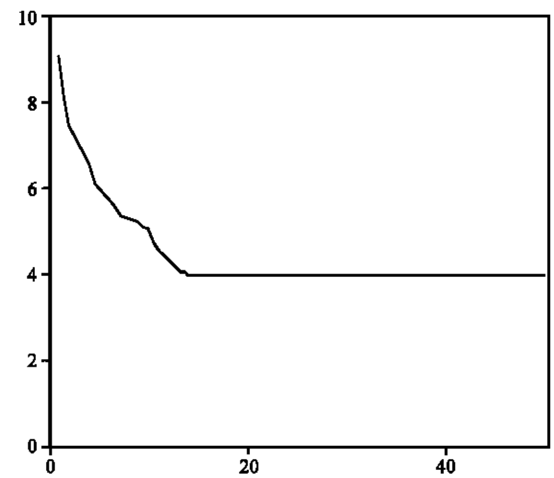

Fig 2: The flow chart of the evaluation of rough set particle swarm

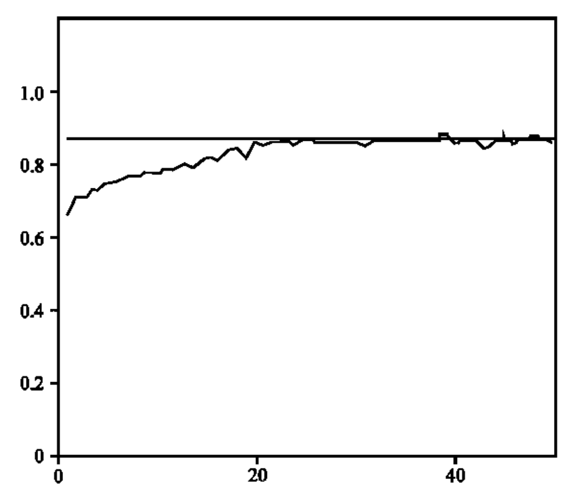

Fig 5: The optimal and the average fitness

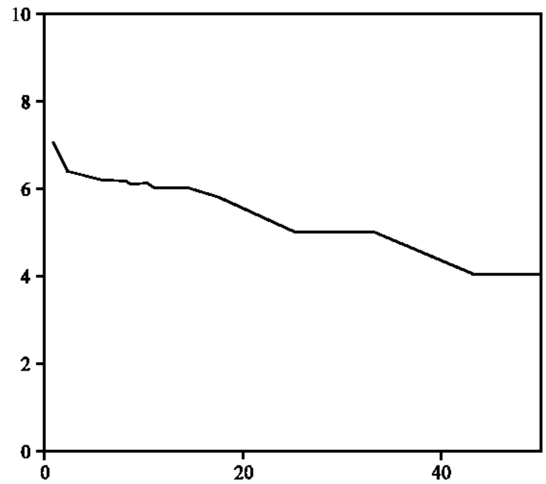

Fig 6: The average number of attributes in optimal solution

reason is that there is inconsistent phenomenon in samples, i.e., there are individual samples whose condition attributes are completely same, but decision attribute is different. This inconsistent phenomenon is mainly due to the two attributes. The optimal result after repeatedly reduction and the maximum dependency of all attributes by directly calculation both equal to 0.8667.

From the results we can conclude that the convergence of average dependency is better in genetic algorithm, while the curve of average fitness fluctuates, but the population is relative stable, the reason of fluctuation is caused by r_3 in equation (1b) which is random. In genetic algorithm, the convergence rate of attribute number in the least reduction is slow, sometimes it cannot converge to the optimal solution when the evolutionary iteration is 50. Therefore, let P_m of mutation operator equals to 0.1 to improve the mutation probability, which can greatly advance the convergence rate in 50 iteration, but influence on the local search of genetic algorithm. We can also see that the convergence rate of rough set particle swarm algorithm is faster.

Rough set particle swarm algorithm and genetic algorithm are respectively operated 50 times, where the termination condition modifies that the attribute number of the minimum reduction is 4, the operation results are shown in Table 3.

The experimental data shows as follows: when = 0.01, the convergence rate of genetic algorithm is very low, the convergence is greatly related with the initial value of population, the convergence rate gradually improves with the maximum iteration of population evolution increased; when P_m = 0.1, the average convergence iteration of genetic algorithm greatly improves and changes the global convergence, but decreases the local search ability; when the population size increases, the average convergence iteration and time of rough set particle swarm decreases, but for genetic algorithm, although the average convergence iteration decreases and the convergence rate increases, the improvement is not obvious, instead the average convergence time increases; it is obviously seen that the convergence time of rough set particle swarm algorithm is less than genetic algorithm and its evolutionary iteration is also less than genetic algorithm, i.e., rough set particle swarm algorithm is better than genetic algorithm.

Table 3: The comparison of two algorithms

Algorithm	Rough set Particle swarm algorithm		Genetic algorithm				
Mutation probability	none		P_m = 0.01			P_m = 0.1	
Maximum iteration	50.0	50.0	50.0	100.0	200.0	50.0	50.0
Population size	20.0	40.0	20.0	20.0	20.0	20.0	40.0
Operation times	50.0	50.0	50.0	50.0	50.0	50.0	50.0
Convergence times	50.0	50.0	5.0	7.0	50.0	48.0	50.0
Average convergence iteration	17.2	14.1	20.0	37.0	160.0	41.8	36.9
Convergence rate (%)	100.0	100.0	10.0	14.0	100.0	96.0	100.0
Average convergence time(second)	7.5	11.4	7.7	16.3	80.3	21.4	38.1

Analysis the running results: because the particle swarm dose not have genetic operation such as crossover and mutation, but searches by its own velocity, so the calculation is comparative simple and it saves time; moreover, the information sharing mechanism of particle swarm is different with the genetic. In genetic algorithm, chromosome shares information with each other, so the population comparative uniformly moves to the optimal area. In the particle swarm, only the global optimal particle in history transfers the information to other particles to guarantee the global convergence of the algorithm, this is the one-way information flow, the local convergence ability is guaranteed by the local optimal particle in history. The whole searching updating process is a process of following current optimal solution and local optimal solution. Compared with genetic algorithm, all particles perhaps quickly converge to optimal solution at most.

Application in pattern recognition: In the pattern recognitions of oil well and dry well, firstly reduce the attributes of sample information, then train and recognize by SVM (Support Vector Machines, Naiyang and Yingjie, 2004). Take Table 2 for example, with the rough set particle swarm algorithm reduced, the reduction result are $\{c_2, c_5, c_8, c_9\}$ and $\{c_2, c_5, c_7, c_9\}$ { Choose$\{c_2,c_5,c_8,c_9\}$ as

Table 4: The comparison of oil trial conclusion

No.	Oil trial conclusion	Accordance
1	Oil	Oil
2	Oil	Oil
3	Oil	Oil
4	Oil	Oil
5	Oil	Oil
6	Oil	Oil
7	Oil	Oil
8	Dry	Dry
9	Dry	Dry
10	Dry	Dry
11	Dry	Dry
12	Dry	Dry
13	Dry	Dry
14	Dry	Dry
15	Dry	Dry
16	Dry	Dry
17	Dry	Dry

Compared with the oil trail conclusion, the accuracy of recognition can reach 100%. This fully indicates that the method of least data reduction in oil reservoir based on rough set particle swarm has a notable effect in application.

example, select No. 31 and 47 as testing samples and recognize the unknown well by them, the result is shown in Table 4.

CONCLUSION

A particle swarm algorithm is presented in this study based on rough set to get the least data reduction and a set of reduction method is designed. This method resolves not only consistent decision table, but also inconsistent. The typical example indicates that it is feasible and is superior to the algorithm of general attribute reduction and the algorithm presented in other references. The practical application in the oil reservoir also shows that the effect is very notable by this method, so it is effective and feasible, especially there are few algorithms about inconsistent decision table. At the same time, the presentation of this method has great actual meaning to further construct attribute reduction algorithm with higher efficiency. We can also further study as follows: Based on the core attribute of decision information table in the particle swarm algorithm, we can introduce the conception of important gene bit in genetic algorithm to improve the operation efficiency.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their careful reading of this paper and for their helpful and constructive comments. This research was supported in part by the National Natural Science Foundation of China under grant no. 60173058.

REFERENCES

Kennedy, J. and R.C. Eberhart, 1995. Particle swarm optimization [A]. Proc. IEEE Int. Conf. Neural Networks [C]. Piscataway, NJ: IEEE Press, 1995, pp: 1942-1948.

Kennedy, J. and R.C. Eberhart, 1997. A Discrete Binary Version of the Particle Swarm Algorithm. In Proceedings of the 1997 conference on Systems, Man and Cybernetics. Piscataway, NJ, IEEE Service Center, pp: 4104-4109.

Liu Hongjie, Feng Boqin, Xia Kewen and Zheng Hongmin, 2006. Seismic Reservoir Oil-Gas Prediction Study Based on Rough Set and RBF Network. 6th World Congress on Intelligent Control and Automation, Conference Proceedings, pp: 4229-4233.

Liu Qing, 2003. Rough Set and Rough Reasoning[M]. Beijing: Science Press.

Li Xiongxi. and Li Jun, 2003. Data Mining and Knowledge Discovery[M]. Beijing: Higher Education Press.

Naiyang., D. and T. Yingjie, 2004. A New Method of Data Mining-Support Vector Machines(the front page). Beijing: Science Press, pp: 1-408.

Pawlak Z., 1982. Rough Sets. Intl. J. Computer Inform. Sci., 11: 341-356.

Pawlak Z., 1999. Rough set theory for intelligent industrial applications. Intelligent Processing and Manufacturing of Materials. IPMM '99. Proceedings of the Second International Conference on 10-15 July 1999, pp: 37-44.

Wong., S.K.M. and W. Ziarko, 1985. On optimal decision rules indecision tables [J]. Bull. Polish Acad. Sci., 33, pp: 693-696.

Zhang Wenxiu and Qiu guofang, 2007. Uncertain decision based on rough set[M]. Beijing: Tsinghua University Press.