A Quick Attribute Reduction Algorithm Based on Incomplete Decision Table

Zhangyan Xu, Jianhua Zhou, and Chenguang Zhang

College of Computer Science & Information Technology Guangxi Normal University, 541004, Guilin, China
xyzwlx72@yahoo.com.cn, zjh1988100@163.com, zhangchenguanga@sohu.com

Abstract. As the object of study incomplete decision table, with the study of the notion of conflict region, the definition of attribution reduction based on conflict region in incomplete decision table is provided. it is proved that the attribute reduction is equivalent to the attribute reduction based on positive region, at the same time ,a new attribute reduction algorithm which is in incomplete decision table is designed, whose time complexity is $O(|K||C|^2|U|)\left(|K|=\max\{|T_C(x_i)|, x_i \in U\}\right)$ .Finally, an example is used to illustrate the efficiency of the new algorithm.

Keywords: rough set, incomplete decision, conflict region, attribution reduction.

1 Introduction

Rough set theory is put forward by Z. Pawlak in 1980s. And it is a mathematical tool which is used to study and deal with the incomplete data and imprecise knowledge. It is used in artificial intelligence and cognitive science, especially in the field of intelligent information processing; it has been widely used [1-4]. The attribution reduction delete irrelevant or redundant attributes in the same classification or decision-making capacity of the Knowledge Base case [5-8].

Some scholars have proved that seeking decision table all attribute reduction and minimum attribute reduction is an NP-Hard problem [9-11]. Many scholars design various of algorithms for attribution reduction in complete decision table. In practical applications due to the measurement error of the data, restrictions of knowledge acquisition and other various reasons, there will be some default values in the decision table. So the decision table that we often have to deal with is incomplete. In general, the tolerance class algorithm of object concentrated two comparison, compare them in each attribute property set whether or not to meet the definition, if meet the tolerance class, belong to the same tolerance class; or to the object of every object, according to whether the value judgment of its property belongs to the tolerance existing class. Reference [4] designs an algorithm for attribution reduction based on positive region in incomplete decision table. Reference [5, 6] gives a method

Y. Yang, M. Ma, and B. Liu (Eds.): ICICA 2013, Part I, CCIS 391, pp. 499–508, 2013.

based on information entropy. The time complexity of both algorithms is $O(|C|^3|U|^2)$.Reference [5] gives a method based on discernibility matrix, and its time complexity is $O(|C|^2|U|^2)$.

This paper combines conflict region with tolerance relation by reference [7-10], and gives a new algorithm for attribution reduction. Finally this algorithm is proved to be correct and efficient.

2 The Basic Concepts of Rough Set

The basic concepts, notations and results of rough sets as well as their extensions are briefly reviewed [12-15].

Definition 1. A decision table could be defined as $S=(U,A,V,f)$ where U is a finite non-empty which is represented the set of objects, $A=C\cup D$ is the attribute set, subset C is called condition attribute and subset D is called decision attribute, $V=\bigcup_{r\in A} V_r$ is the set of attribute values, $f:U\times R\to V$ is an information function, which offers an attribute value to each attribute of each object, if $\forall r\in C\bigcup D,\ x\in U$, then $f(x,r)\in V_r$.

If there exists at least $a\in C$ and V_a contains uncertain value (marked $f(x,a)=*$), then the decision table is called incomplete decision table.

Definition 2.[3, 16]. Suppose T is a binary relation of Incomplete Information System $S=(U,A,\{V_a,a\in A\},\{f_a,a\in A\})$, and T is defined as follow: $\forall x,y\in U$, $\forall B\in R$, $T(x,y)\Leftrightarrow(\forall b\in B)((b(x)=b(y))\vee(b(x)=*)\vee(b(y)=*))$.

Definition 3 [3]. $T_B(x)$ is the tolerance class of object x which denotes that object set meets the tolerance relation with x on condition attribute set B: $T_B(x)=\{y \mid y\in U\wedge T(x,y)\}$.The upper and lower approximations of X with regard to B under the characteristic relation are $X^B=\{x\mid x\in U\wedge I_B(x)\cap X\neq\varnothing\}$, $X_B=\{x\mid x\in U\wedge I_B(x)\subseteq X\}$.

Definition 4. Let $S=(U,A,V,f)$ be an incomplete decision table, $A=C\bigcup D$. If $Q\subseteq U$, $P\subseteq C$, the positive region of Q with respect to P is defined as follow: $POS_P(\mathrm{Q})=\bigcup\{x\mid T_P(x)\subseteq Y\}$, $Y\in Q/D$ .if $P=C$, $Q=U$,then $POS_C(D)=\bigcup\{x\mid T_P(x)\subseteq D_i\}\quad D_i\in U/D$.

Definition 5. Let $S=(U,A,V,f)$ be an incomplete decision table, $A=C\bigcup D$, $P\subseteq Q\subset C$, $U/P=\{T_P(x_1),T_P(x_2),\cdots\ T_P(x_{|U|})\}$, $U/Q=\{T_Q(x_1),T_Q(x_2),\ldots T_Q(x_{|U|})\}$, $\forall T_P(x_i)\in U/P\Rightarrow\exists T_Q(x_i)\in U/Q$,then $T_Q(x_i)\subseteq T_P(x_i)$

Definition 6. Let $S=(U,A,V,f)$ be an incomplete decision table, $A=C\bigcup D$, $\forall a\in B\subseteq C$,if $POS_B(D)$=$POS_{B-\{a\}}(D)$,a is unnecessary for B relative to D, Otherwise a is necessary for B. if arbitrary element of B is unnecessary, we call that B is independent with respect to D.

3 The Algorithm of Computing Conflict Region

In the section, we give the definition of conflict at first, and then we design an algorithm of computing conflict region in incomplete decision table [16-20].

As the object of study incomplete decision table, with the study of the notion of conflict region, the definition of attribution reduction based on conflict region in incomplete decision table is provided.

Many scholars design several of algorithms for attribution reduction in complete decision table [21-23]. In practical applications due to the measurement error of the data, restrictions of knowledge acquisition and other various reasons, there will be some default values in the decision table. So the decision table that we often have to deal with is incomplete.

In general, the tolerance class algorithm of object concentrated two comparison, compare them in each attribute property set whether or not to meet the definition, if meet the tolerance class, belong to the same tolerance class; or to the object of every object, according to whether the value judgment of its property belongs to the tolerance existing class.

Definition 7. Let $S=(U,A,V,f)$ be an incomplete decision table, $A=C\bigcup D$, $B\subseteq C$ .conflict region $Conset(B)$ is defined as: $Conset(B)$={x_i|$x_i\in U$,$\exists$$x_m$,$x_n$$(x_m,x_n)$ $\in T_B(x_i)\wedge f(x_m,D)\neq f(x_n,D)\}$.

Algorithm 0[8].Finding tolerance class $T_P(x)$

a) Input: An incomplete decision table S=(U,C,D,V,f) , $U=\{x_1,x_2,\cdots x_n\}$, $c_i\in C$;

Output: $T_{c_i}(x)$

Step1. for $a_i\in C$ statistics maximum value、 minimum value and if there has default value("*") of $f(x_j,c_i)$ (i=1,2,⋯,|U|),each mark M_i, M_i, c_i^*;// $c_i^*=1$ note exist default value, $c_i^*=0$ note not exist default value;

Step2:distribution:: if $c_i^*=1$ note exist default value, establish Empty queue of $A=M_i-m_i+2$, or establish Empty queue of $A=M_i-m_i+1$, and each mark the $front_k$ and end_k ($k=0,1,2,\cdots,M_i-m_i\ or\ M_i-m_i+1$) link the front point and the rear point of the k'th queue, $front_*$ and end_* each link the front point and the rear point of the last queue ("*"queue) when $c_i^*=1$ .we can put the element of the list into the $f(x,a_i)$-m_i queue or put into the last queue("*"queue);

Step3.:collection:the header of list link the first not empty queue , and then mortify the each rear point of the not empty queue , link the next front point of the not empty queue , finally make the all queue one list;

Step4. Suppose the object sequence of the list from the step3 note $x_1^{'}, x_2^{'}, \cdots, x_n^{'}$;

t=1; $E_1 = \{x_1^{'}\}$;

$for(j = 2; j <= n; j++)$

$if\,(f\,(x_j^{'}, a_i) = f\,(x_{j\text{-}1}^{'}, a_i))$

$E_t = E_t \cup \{x_j^{'}\}$;

$else\{$t=t+1;$E_t = E_t \cup \{x_j^{'}\};\}$

Step5. $if\,(c_i^* = 1)$

for(v = 1;v<t;v++)

$\{T_v = E_v \cup E_t$;

$T_t = E_t \cup E_v\,(v = 1, 2, \cdots, t\text{-}1);\}$

$else$

$T_v = E_v$;

The algorithm completes the first time division based on one of the attributes, The algorithm learns from the thought of equivalence partitioning which work in complete decision table.

b) Algorithm of positive region

Step1. $U_{pos} = \varnothing, U_{neg} = \varnothing, T_{C \cup \{c_i\}}$;

Step2.for （j=1;j<r;j++）

$if\ \ (f(x, c_j) = *)\ T_{C \cup \{c_j\}}(x) = T_C(x)$;

else

$\forall y \in T_C(x)$;

$if\,(f(y, c_j) == * \vee f(y, c_j) == f(x, c_j))$

$T_{C \cup \{c_j\}}(x) = T_{C \cup \{c_j\}}(x) \bigcup \{y\}$;

Step3.if j>r, then goto Step4

else $C = C \bigcup \{c_j\}$, goto Step2;

Step4. Output $T_C(x)$.

By adding a partition condition number, each equivalence class made corresponding change. At last, all conditions are joined the attribution reduction set, and we get the final equivalence class.

As the object of study incomplete decision table, with the study of the notion of conflict region, the definition of attribution reduction based on conflict region in incomplete decision table is provided.

Algorithm 1. Finding the conflict region $Conset(P)$ with respect to P .

Input : An incomplete decision table $S=(U,A,V,f)$, $A=C \cup D$, $\varnothing \neq P \subseteq C$;

Output: $Conset(P)$;

Step 1. Initialize $Conset(P)=\varnothing$;

Step 2. Use the algorithm 1 and 2 from reference [8] to find the tolerance class $T_P(x)$

Step 3. $for(i=1;i \leq |U|;i++)$

$if(|T_P(x_i)|>1)$

$if(\forall x_m, x_n \in T_P(x_i)$, $f(x_m,D) \neq f(x_n,D))$;

$Conset(P)=Conset(P) \cup \{x_i\}$;

Step 4. Output $Conset(P)$;

4 New Attribute Reduction Algorithm

In the section, we will introduce how to compute attribute reduction based on conflict region, and prove that it is equivalent to the algorithm based on positive region.

Theorem 1. Let $S=(U,A,V,f)$ be an incomplete decision table, $A=C \bigcup D$, $P \subseteq Q \subseteq C$, then $|conset(P)| \geq |conset(Q)|$, $|conset(P)|$ is the number of set $conset(P)$.

Proof. We just need to proof $conset(Q) \subseteq conset(P)$, $\forall x_i \in conset(Q)$,we know that there are two objects decision values are not equal in $T_Q(x_i)$, then $\exists x_i \in T_P(x_i)$ and $T_Q(x_i) \subseteq T_P(x_i)$ based on definition 5, so there are two objects decision values are not equal in $T_P(x_i)$ .Due to the arbitrariness of the x_i , we know the result is true.

Theorem 2. Let $S=(U,C \cup D,V,f)$ be an incomplete decision table, $R \subseteq C$, R is the attribute reduction of S ,if and only if $|ConSet(R)|=|ConSet(C)|$.

Proof. We just need to prove $|ConSet(R)|=|ConSet(C)| \Leftrightarrow POS_R(D)=POS_C(D)$. First we prove $POS_R(D)=POS_C(D) \Rightarrow |ConSet(R)|=|ConSet(C)|$. $POS_R(D)=POS_C(D)$ We can know $|POS_R(D)|=|POS_C(D)|$, and $|U-POS_R(D)|=|U-POS_C(D)|$, so the result is $|ConSet(R)|=|ConSet(C)|$.

Second we prove $|ConSet(R)|=|ConSet(C)| \Rightarrow POS_R(D)=POS_C(D)$, Due to $R \subseteq C$, then $POS_R(D) \subseteq POS_C(D)$ if $POS_R(D) \supseteq POS_C(D)$ is not established, then $POS_R(D) \subset POS_C(D)$ .so $\exists x_0 \in U$, $x_0 \in POS_C(D)$ and $x_0 \notin POS_R(D)$ .then we have $x_0 \in U-POS_R(D)$,but $x_0 \notin U-POS_C(D)$,so $|U-POS_R(D)| \neq |U-POS_C(D)|$.

However $POS_R(D) \subset POS_C(D)$, thus $|POS_R(D)| < |POS_C(D)|$, so $|ConSet(R)| > |ConSet(C)|$. So $POS_R(D) \supseteq POS_C(D)$ is true.

Altogether the result is true.

Theorem 3. The attribute reduction based on conflict region is equivalent to the attribute reduction based on positive region.

Proof. The theorem 3 is easy to be proved true based on the theorem 2.so we don't prove anymore.

Definition 8. Let $S=(U,A,V,f)$ be an incomplete decision table, $A=C\bigcup D$, $R\subset C$, $a\in C-R$,the significance of attribute a is defined as $Sig(a,R,D)=|ConSet(R)| -|ConSet(R\cup\{a\})|$.

From definition 8 we can see that the important degree of attribute a is bigger and the conflict region reject can be separate faster. So the algorithm can speed up the convergence.

Algorithm 2. attribute reduction algorithm based on conflict region

Input： An incomplete decision table $S=(U,A,V,f)$, $A=C\cup D$, $U=\{x_1,x_2,\cdots x_n\}$, $C=\{c_1,c_2,\cdots c_r\}$;

Output： Attribute reduction R

Step 1.we can compute $conset(C)$ based on Algorithm 1. Initialize $R=\varnothing$, $conset(\varnothing)=U$;

Step 2. $while(|ConSet(R)|\neq|ConSet(C)|)$

$for(i=1;i<=r;i++)$

compute $conset(R\cup c_i)$ and choose the c_i of the largest of $sig(c_i,R,D)$;

$R=R\cup c_i$;

Update $conset(R)$;

Step 3. $for(i=1;i<=|R|;i++)$

$R^{'}=R-c_i$;

$if(|conset(R^{'})|=|conset(R)|)$

$R=R^{'}$;

Step 4.output R;

In Algorithm 2, based on the reference [8], we know the time complexity of Step 1 is $O(|K||C||U|)$ $K=\max\{|T_C(x_i)|,x_i\in U\}$. In the worst case, Step 2 need circle $|C|$ times, so its time complexity is $O(|K||C|^2|U|)$. Step 3 is same to Step 2. Therefore, the total time complexity of algorithm 2 is $O(|K||C|^2|U|)$, $K=\max\{|T_C(x_i)|,x_i\in U\}$.

5 Case Analysis

We use an example to illustrate that the algorithm, and the example (table 1) is as follow:

Table 1 is a incomplete decision table, in order to facilitate the description of the algorithm steps, the data set we used has only 8 objects. However the relationship between these 8 objects covers almost the entire situation. ("*"is marked as the default value). In general, the tolerance class algorithm of object concentrated two comparison, compare them in each attribute property set whether or not to meet the definition, if meet the tolerance class, belong to the same tolerance class; or to the object of every object, according to whether the value judgment of its property belongs to the tolerance existing class. Through calculation examples, learn how to improve the efficiency of classification. As the object of study incomplete decision table, with the study of the notion of conflict region, the definition of attribution reduction based on conflict region in incomplete decision table is provided.

Table 1. Incomplete decision table

U	a1	a2	a3	a4	a5	D
x1	1	1	2	1	*	1
x2	2	*	2	1	1	1
x3	*	*	1	2	2	2
x4	1	*	2	2	1	1
x5	*	*	2	2	1	3
x6	2	1	2	*	1	1
x7	1	1	2	1	2	1
x8	2	*	2	1	*	1

Step 1. we get $conset(C) = \{x_4, x_5, x_6\}$ based on Algorithm 1.

Step 2. Due to $R = \varnothing$, $conset(\varnothing) = U$, then $conset(R) = U$, $|ConSet(R)| \neq |ConSet(C)|$ .So go into the while circle, in the first circle , add attribution a1 $conset(a_1) = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$, $sig(a_1, R, D) = 0$; add attribution a2 $conset(a_2) = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$, $sig(a_2, R, D) = 0$; add attribution a3 $conset(a_3) = \{x_1, x_2, x_4, x_5, x_6, x_7, x_8\}$, $sig(a_3, R, D) = 1$; $conset(a_4) = \{x_3, x_4, x_5, x_6\}$, $sig(a_4, R, D) = 4$; we choose attribute a_4 , $R = \{a_4\}$. because $|ConSet(R)| \neq |ConSet(C)|$, we continue to increase attribute. $conset(a_4, a_1) = \{x_3, x_4, x_5, x_6\}$, $sig(a_1, R, D) = 0$; $conset(a_4, a_2) = \{x_3, x_4, x_5, x_6\}$, $sig(a_2, R, D) = 0$; $conset(a_4, a_3) = \{x_4, x_5, x_6\}$, $sig(a_3, R, D) = 1$; we choose attribute a_3 , $R = \{a_4, a_3\}$ .because of $|ConSet(R)| = |ConSet(C)|$,and then the Algorithm is end.

Step 3 attribution a_4, a_3 are calculated can not be deleted, and $R = \{a_4, a_3\}$ is the result.

After verification the result is same to the algorithm based on the positive region. It is proved that the attribute reduction is equivalent to the attribute reduction based on positive region.

6 Experimental Comparison

We use 6 data sets in the UCI database as the experimental object, and then compare with the algorithm in the reference [10]. The running time of reference [10] is marked as T1, and the running time of this paper is marked as T2. The result is as follows. In order to enhance the reliability of experimental results, the average time of 5 experiments is taken as the final time in this paper.

As the object of study incomplete decision table, with the study of the notion of conflict region, the definition of attribution reduction based on conflict region in incomplete decision table is provided. In the follow table we just give the name of data sets, the final attribute reduction and the running time of the two algorithms. For rigorous this paper has described the detailed experiment environment. The detailed data is as follows.

In this experiment, Computer hardware configuration is as follows: Intel Pentium(R) D 3.20 GHZ, memory: 1G, Development platform: VS2008. The detailed experiment data is as follows:

Table 2. Experimental result

Data set	Attribute reduction	T1(ms)	T2(ms)
Credit	2,3,8,6	14271.30	10370.20
Car	2,1,4,6,5,3	40230.62	12630.55
Hepatitis	1,16,6,3,5	1840.365	735.382
Soybean-large	17,7,16,12,1,22,6,15,8,35,4,9	13570.75	10260.65
Vote	9,4,3,13,1,2,11,10,12,16,15	3190.625	2169.535
win	2	510.125	387.532

The second column is described as the number of the attribution in the database. These data sets include complete and incomplete. To verify the efficiency of the algorithm we use the different kinds of data sets. As can be watched from the experimental data in the table, in all experiments the running time of this paper are faster than it in reference [10]. Based on the different experimental environment, there are some partial data that are not same with reference [10]; however, the result does not affect the efficiency of the algorithm. This algorithm is proved to be correct and efficient.

The running time of the program is almost always not the same, this paper repeat testing several times and method the mean value to ensure the reliability of data. In order to get accurate data, this paper use MS as a unit, and retained two decimal places. At last, we proved that the experimental results demonstrate that the algorithm is effective.

7 Conclusion

Attribute reduction is one of the important research content of rough set theory. Equivalence partitioning is one of the difficulties. In general, the tolerance class algorithm of object concentrated two comparison, compare them in each attribute property set whether or not to meet the definition, if meet the tolerance class, belong to the same tolerance class; or to the object of every object, according to whether the value judgment of its property belongs to the tolerance existing class. As the object of study incomplete decision table, with the study of the notion of conflict region, the definition of attribution reduction based on conflict region in incomplete decision table is provided.

These paper researches some attribute reduction algorithms in incomplete decision table, and find their time complexity are not better. Then a new attribute reduction algorithm is designed which based on conflict region, its time complexity is $O(|K||C|^2|U|)$, $K = \max\{|T_C(x_i)|, x_i \in U\}$.

then it is proved that the attribute reduction is equivalent to the attribute reduction based on positive region. At last, an example is used to prove the algorithm is correct and efficient.

Acknowledgements. This work was supported by grant: No.60963008 from the National Natural Science Foundation of China and grant: No. 2011GXNSFA018163 from the Natural Science Foundation of Guangxi, and the Innovation Project of Guangxi Graduate Education.

References

1. Pawlak, Z., Skowron, A.: Rudiments of rough sets. Information Science 117, 3–37 (2007)
2. Pawlak, Z., Wong, S.K.M., Ziarko, W.: Rough sets: probabilistic versus deterministic approach. Computational Intelligence 29, 81–95 (1988)
3. Hu, F., Huang, H., Wang, G.Y.: Granular Computing in Incomplete Information Systems. Mini-Micro Systems 26, 1335–1339 (2005)
4. He, W., Liu, C.Y., Zhao, J.: An algorithm of Attributes Reduction in Incomplete Information System. Computer Science 31, 117–119 (2004)
5. Shu, W.H., Xu, Z.Y., Qian, W.B.: Attribution reduction algorithm based on discernibility matrix of incomplete decision table. Computer Engineering and Applications 47, 105–110 (2011)
6. Huang, B., Zhou, X.Z., Zhang, R.R.: Attribution reduction algorithm based on information content in incomplete information Systems. Systems Engineering Theory and Practice 25, 55–60 (2005)
7. Ge, H., Li, L.S., Yang, C.J.: An Efficient Attribute Reduction Algorithm Based on Conflict Region. Chinese Journal of Computers 35, 342–350 (2012)

8. Shu, W.H., Xu, Z.Y., Qian, W.B.: Quick Attribution Reduction Algorithm Based on Incomplete Decision Table. Journal of Chinese Computer Systems 32, 1867–1871 (2011)
9. Wong, S.K.M., Ziarko, W.: Optimal decision rules in decision table. Bulletin of Polish Academy of Sciences 33, 693–696 (1985)
10. Wang, W., Xu, Z.Y., Li, X.Y.: Attribute reduction algorithm based on object matrix in incomplete decision table. Computer Science 39, 201–204 (2012)
11. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences 11, 341–356 (1982)
12. Skowron, A., Rauszer, C.: The Discernibility Functions Matrics and Functions in Information Systems. In: Slowinski, R. (ed.) Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory, pp. 331–362. Kluwer Academic Publisher, Dordrecht (1992)
13. Hu, X.H., Cercone, N.: Learning in Relational Database: A Rough Set Approach. International Journal of Computional Intelligence 11(2), 323–338 (1995)
14. Nguyen, H.S., Nguyen, S.H.: Some Efficient Algorithms for Rough Set Methods. In: The Sixth International Conference, Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 1996), Granada, Spain, July 1-5, vol. 2, pp. 1451–1456 (1996)
15. Wang, G.Y., Yu, H., Yang, D.C.: Decision Table Reduction Based on Conditional Information Entropy. Chinese Journal of Computers 25(7), 759–766 (2002)
16. Liu, S.H., Cheng, Q.J., Shi, Z.Z.: A New Method for Fast Computing Positve Region. Journal of Computer Research and Development 40(5), 637–642 (2003)
17. Wang, J., Wang, J.: Reduction Algorithms Based on Discernibility Matrix: the Ordered Attributed Method. Journal of Computer Science and Technology 11(6), 489–504 (2001)
18. Zhao, M., Wang, J.: The Data Description Based on Reduct. PhD Thesis, Institute of Automation, Chinese Academy of Sciences, Beijing, China (2004) (in Chinese)
19. Moshkov, M.J., Piliszczuk, M., Zielosko, B.: On Partial Covers, Reducts and Decision Rules with Weights. In: Peters, J.F., Skowron, A., Düntsch, I., Grzymała-Busse, J.W., Orłowska, E., Polkowski, L. (eds.) Transactions on Rough Sets VI. LNCS, vol. 4374, pp. 211–246. Springer, Heidelberg (2007)
20. Qin, Z.R., Wu, Y., Wang, G.Y.: A Partition Algorithm for Huge Data Sets Based on Rough Set. Pattern Recognition and Artificial Intelligence 19(2), 249–256 (2006)
21. Hu, F., Wang, G.Y.: Analysis of the Complexity of Quick Sort for Two Dimension Table. Chinese Journal of Computers 30(6), 963–968 (2007) (in Chinese)
22. Hu, F., Wang, G., Xia, Y.: Attribute Core Computation Based on Divide and Conquer Method. In: Kryszkiewicz, M., Peters, J.F., Rybiński, H., Skowron, A. (eds.) RSEISP 2007. LNCS (LNAI), vol. 4585, pp. 310–319. Springer, Heidelberg (2007)
23. Wang, G.Y.: Rough Set Theory and Knowledge Acquisition. Xi'an Jiaotong University Press, Xi'an (2001) (in Chinese)