Journal of Network and Innovative Computing
ISSN 2160-2174 Volume 2 (2014) pp. 196-204

Dynamic Attribute Reduction Algorithms in Set-valued Decision Systems by Rough Set

Thi Thu Hien Phung[1], Van Tho Ninh[2]

[1] University of Economic and Technical Industries
456 Minh Khai, Hai Ba Trung District, Ha Noi, Viet Nam
Thuhiencn1@gmail.com

[2] University of Economic and Technical Industries
456 Minh Khai, Hai Ba Trung District, Ha Noi, Viet Nam
Thodtu.nd@gmail.com

***Abstract*: Attribute reduction is an important research concept in rough set theory. However, most of attribute reduction methods are performed on single-valued decision system decision table. In this paper, the author's methods for attribute reduction in static set-valued decision systems and dynamic set-valued decision systems with dynamically increasing and decreasing conditional attributes. The methods use generalized discernibility matrix and function in tolerance-based rough sets.**

Keywords: Tolerance relations, rough sets, set-valued decision system, generalized discernibility matrix, attribute reduction, decision table.

I. Introduction

The theory of conventional rough set initiated by Pawlak [5] is an effective tool to solve attribute reduction problems and to extract rules in information systems. In real information systems, attribute value of object might be a set. For example, consider an information system which has a target "Nguyen Van A" with the attributions "foreign languages" containing "English, French, Russian"; that is Nguyen Van A can speak English, or French, or Russian. Such information system is called set-valued information system.

Attribute reduction in decision systems is the process of choosing the minimum set of the conditional attribute set, preserving classified information of the decision systems. In decision systems, computer scientists have provided several attribute reduction methods based on model of conventional rough set, summarized by Shifei D et. al. in ref. [8]. In set-valued information system, Guan Y. Y. Wang et. al. [2] expanded equivalent relation in conventional rough set to tolerance relation and developed model tolerance-based rough set by expanding lower approximation, upper approximation, positive domain, etc. based on tolerance relation. There are remarkable reports about attribute reduction in decision system and ordered decision system in model of tolerance-based rough set approach in ref. [1, 6, 10]. In ref. [12], the authors using matrix method studied the altering of approximation sets with and without attribute set.

However, studies at attribute reduction when decision table of set value varies (with/without attribute set, with/without target set) need to be more developed. With the ideas of discernibility matrix and discernibility function in theory of conventional rough set initiated by Skowron [9], in this paper, the author develop generalized discernibility matrix and generalized discernibility function. Utilizing generalized discernibility; developing an attribute reduction method in two cases: decision system of set value does not vary and does vary with or without attribute set.

In this paper, section 2 describes the results of set-value decision system and definitions of reduct. In section 3, the author demonstrate attribute reduction method using generalized discernibility functions. In section 4, the author provide attribution reduction method in case of adding and deleting of an attribute set. In section 5, the author provide relation between.

II. Basic Definitions

In this part, the author present some basic definitions about set-valued information system in [2, 12].

Information systems is a tuple $IS = (U, AT)$, where U is a finite set of objects and AT is a finite set of attribute. The value of an attribute $a \in AT$ at an object $u \in U$ is denoted as a(u), the value of an attribute $a \in AT$ at an object $v \in U$ is denoted as a(v). Each sub-set $A \subseteq AT$ determines one equivalence relation:

$$IND(A) = \{(u,v) \in U \times U \mid \forall a \in A,\ a(u) = a(v)\}$$

Partition of U generated by a relation IND(A) is denoted as U/A and equivalence class in the partition U/A which includes $u \in U$ is denoted as $[u]_A$, while $[u]_A = \{v \in U \mid (u,v) \in IND(A)\}$. It is easy to see that $[u]_A = \cap [u]_{\{a\}}$ with all $a \in A$.

Considering information system IS = (U, AT), if existing $u \in U$ in order to $a(u)$ includes at least two values, and then $IS = (U, AT)$ is called set-valued information system.

A set-valued decision information system $DS = (U, AT \cup \{d\})$ is a special case of a set-valued information system, where U is a non-empty finite set of objects, AT is a non-empty finite set of condition attribute and d is a decision attribute with $AT \cap \{d\} = \varnothing$; $V = V_{AT} \cup V_d$, where V_{AT} is the set of condition attribute values and V_d is the set of decision attribute values; f is a mapping from $U \times (AT \cup \{d\})$ to V such that $f : U \times AT \to 2^{V_{AT}}$ is a set-valued mapping and $f : U \times \{d\} \to V_d$ is single-valued mapping.

In the set-valued information system (U, AT), for $b \in A$, the tolerance relation T_b is defined as:

$$T_b = \{(u,v) \mid f(u,b) \cap f(v,b) \neq \varnothing\}.$$

and for $B \subseteq A$, the tolerance relation T_B is defined as follows:

$$T_B = \{(u,v) \mid \forall b \in B, f(u,b) \cap f(v,b) \neq \varnothing\} = \bigcap_{b \in B} T_b.$$

The generalized decision in set-valued decision system is similar to incomplete decision system [3]. Let $DS = (U, AT \cup \{d\})$ be a set-valued decision system, for $u \in U$, $\partial_{AT}(u) = \{d(v) \mid v \in T_{AT}(u)\}$ is called generalized decision of u on attribute set AT. If $|\partial_{AT}(u)| = 1$ for all $u \in U$ then DS is consistent, otherwise it is inconsistent.

Similarity, for incomplete decision systems [3], reduct of set-valued decision system is defined as follows:

Definition 1. Let $DS = (U, AT \cup \{d\})$ be a set-valued decision system. If $R \subseteq AT$ satisfies

(1) $\partial_R(u) = \partial_{AT}(u)$ for any $u \in U$.

(2) For $\forall R' \subset R$, there exist $u \in U$ such that $\partial_{R'}(u) \neq \partial_{AT}(u)$. Then R is called a reduct of DS based on generalized decision system.

Example 1. Considering set-valued decision system $DS = (U, AT \cup \{d\})$ in Table 1

Table 1. Set-valued decision system

U	a_1	a_2	a_3	a_4	*d*
u_1	{1}	{1}	{1}	{0}	1
u_2	{0}	{0, 1}	{1}	{0}	1
u_3	{0, 1}	{0, 1}	{0}	{1}	0
u_4	{1}	{0, 1}	{1}	{1}	1
u_5	{0, 1}	{0, 1}	{1}	{1}	2
u_6	{0}	{1}	{1}	{0, 1}	1

For $u_1 \in U$ we have $T_{a_1}(u_1) = \{u_1, u_3, u_4, u_5\}$, $T_{a_2}(u_1) = U$, $T_{a_3}(u_1) = \{u_1, u_2, u_4, u_5, u_6\}$, $T_{a_4}(u_1) = \{u_1, u_2, u_6\}$. So $T_{AT}(u_1) = T_{a_1}(u_1) \cap T_{a_2}(u_1) \cap T_{a_3}(u_1) \cap T_{a_4}(u_1) = \{u_1\}$.

Similarity, $T_{AT}(u_2) = \{u_2, u_6\}$, $T_{AT}(u_3) = \{u_3\}$, $T_{AT}(u_4) = \{u_4, u_5\}$, $T_{AT}(u_5) = \{u_4, u_5, u_6\}$, $T_{AT}(u_6) = \{u_2, u_5, u_6\}$.

In addition, $\partial_{AT}(u_1) = \partial_{AT}(u_2) = \{1\}$, $\partial_{AT}(u_3) = \{0\}$, $\partial_{AT}(u_4) = \partial_{AT}(u_5) = \partial_{AT}(u_6) = \{1, 2\}$.

Thus, DS is inconsistent.

Definition 2. For set-valued decision system $DS = (U, AT \cup \{d\})$. If $R \subseteq AT$ satisfies:

(1) $POS_R(\{d\}) = POS_{AT}(\{d\})$

(2) $\forall R' \subset R$, $POS_{R'}(\{d\}) \neq POS_{AT}(\{d\})$.

then *R* is called as a *reduct of DS based on positive region*.

III. Generalized discernibility function based attribute reduction in set-valued decision system

Attribute reduction in decision system is a process of selecting the minimal sub-set of conditional attribute set, preserving classified information of the decision systems. In traditional rough set, Skowron [9] proposed discernibility matrix and discernibility function to find reduct. Based on this approach, the author propose generalized discernibility matrix and generalized discernibility function in order to find a reduct in set-valued decision system.

Generalized discernibility matrix is constructed based on discernibility matrix [9]. Elements in Generalized discernibility matrix is 0 or 1. Elements 1 is denoted as a pair of objects discerned by conditional attribute set with respect to generalized decision [3].

Definition 3. Let $DS = (U, AT \cup \{d\})$ be a set-valued decision system, $A \subseteq AT$ and $|U| = n$. The generalized discernibility matrix $M_A = (m_{ij})_{n \times n}$ of the DS, is square matrix, each element has a value of 0 or 1, is defined as follows:

(1) $m_{ij} = 1$ if $d(u_j) \notin \partial_A(u_i)$

(2) $m_{ij} = 0$ if $d(u_j) \in \partial_A(u_i)$.

Note: If $A = \varnothing$ so stipulated that $m_{ij} = 0$ and M_A is not symmetric matrix because of $d(u_j) \notin \partial_A(u_i)$ still has $d(u_i) \in \partial_A(u_j)$ with $i = 1,...,n$; $j = 1,...,n$.

*Example 2***.** With set-valued decision system in the example 1, generalized discernibility matrix of DS in attribution set AT as follow:

$$M_{AT} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Definition 4. For two matrices $X = (x_{ij})_{m \times n}$ and $Y = (y_{ij})_{m \times n}$, preference relations "$\preceq$" and "$\succeq$" are defined as followed:

(1) $X \preceq Y$ if and only if $x_{ij} \leq y_{ij}$, $i = 1,2,...,m$, $j = 1,2,...,n$.

(2) $X \succeq Y$ if and only if $x_{ij} \geq y_{ij}$, $i = 1,2,...,m$, $j = 1,2,...,n$.

Proposition 1. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system, $P,Q\subseteq AT$.

If $P\subseteq Q$ then $M_P \preceq M_Q$.

Example 3. Continue with example 2, assuming $A\subseteq AT$ with $A=\{a_1,a_2,a_3\}$, got:

$$M_A=\begin{bmatrix}0&0&1&0&0&0\\0&0&1&0&0&0\\1&1&0&1&1&1\\0&0&1&0&0&0\\0&0&1&0&0&0\\0&0&1&0&0&0\end{bmatrix}$$

Obvious, $M_A \preceq M_{AT}$.

Definition 5. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system, $A\subseteq AT$ and $M_A=(m_{ij})_{n\times n}$ is generalized discernibility matrix of DS. Generalized

discernibility function DIS(A) is defined as:

$$DIS(A)=\sum_{i=1}^{n}\sum_{j=1}^{n}m_{ij}\ ,\ 1\le i\le n, 1\le j\le n\ .$$

Generalized discernibility function denotes the number of pair of objects discerned by conditional attribute set in set-valued decision tables.

Example 4. Continue of Example 2, M_{AT} is generalized discernibility matrix, generalized discernibility function is:

$$DIS(AT)=2+2+5+1+1+1=12.$$

From definition 5 and Proposition 1 we have the following proposition:

Proposition 2. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system, $P,Q\subseteq AT$.

If $P\subseteq Q$ then $DIS(Q)\ge DIS(P)$.

Proposition 3. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system and M_{AT}, DIS(AT) corresponding is generalized discernibility matrix and generalized discernibility function of DS in attribution set AT. Then, $DIS(R)=DIS(AT)$ if and only if $\partial_R(u)=\partial_{AT}(u)$ for $\forall u\in U$.

Proof : i) Assume that there exists $u_{i_0}\in U$ such that $\partial_R(u_{i_0})\ne\partial_{AT}(u_{i_0})$. Because $\partial_{AT}(u_{i_0})\subseteq\partial_R(u_{i_0})$ so there exists d_{j_0} in order to $d_{j_0}\in\partial_R(u_{i_0})\wedge d_{j_0}\notin\partial_{AT}(u_{i_0})$.

From $d(u_{j_0})\notin\partial_{AT}(u_{i_0})$ we have $m_{i_0j_0}=1$, $m_{i_0j_0}\in M_{AT}$ (1). From $m_{i_0j_0}=0$, $m_{i_0j_0}\in M_R$ (2).

From assumption $R\subseteq AT$ we have $M_R\preceq M_{AT}$, combined with (1) and (2) we have $DIS(R)\ne DIS(AT)$, that contradicting the condition $DIS(R)=DIS(AT)$. So, if $DIS(R)=DIS(AT)$ then $\partial_R(u)=\partial_{AT}(u)$.

ii) In opposite site, suppose that $DIS(R)\ne DIS(AT)$. According Proposition 1, from $R\subseteq AT$ we have $M_R\preceq M_{AT}$, combining with $DIS(R)\ne DIS(AT)$ we have $M_R\ne M_{AT}$, that existing i_0 and j_0 in such a way as to $m_{i_0j_0}\in M_R, m_{i_0j_0}=0$ (3) and $m_{i_0j_0}\in M_{AT}, m_{i_0j_0}=1$ (4). From (4), infering $d(u_{j_0})\notin\partial_{AT}(u_{i_0})$. From (3) we have $d(u_{j_0})\in\partial_R(u_{i_0})$. So, $\partial_R(u_{i_0})\ne\partial_{AT}(u_{i_0})$, that contradicting condition $\partial_R(u)=\partial_{AT}(u)$ $(\forall u\in U)$. After that if $\partial_R(u)=\partial_{AT}(u)$ with $\forall u\in U$ then $DIS(R)=DIS(AT)$.

From i) and ii) we got the proof

Next, the author will present a method of generalized discernibility function based attribute reduction in set-valued decision system. Our method is similar to other methods of attribute reduction in the traditional rough set, the method consists of the definition of reduct, the difinition of attribute importante and a heuristic algorithm to fine the best reduct based on attribute importance. Definition 5 showed that generalized indiscernibility function DIS(A) specifies for ability to classify of $A\subseteq AT$ in each level which created by the attribution d. Thus, in order to find reduct, the author used generalized discernibility function as standard to selecting heuristic algorithms, it called an importance of attribution.

Definition 6. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system. If $R\subseteq AT$ satisfies:

(1) $DIS(R)=DIS(AT)$.

(2) $\forall R'\subset R$, $DIS(R')\ne DIS(AT)$. Then R is called a reduct of DS based on generalized discernibility function.

Proposition 3 proved that generalized discernibility function based reduct which is similarity with reduct based generalized decision system.

Definition 7. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system, $A\subset AT$ and $a\in AT-A$. The importance of attribute a for attribute set A is defined as:

$$SIG_A^{out}(a)=DIS(A\cup\{a\})-DIS(A).$$

Definition 8. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system, $A\subset AT$ and $a\in A$. The importance of attribute a in attribute set A is defined as:

$$SIG_A^{in}(a)=DIS(A)-DIS(A-\{a\}).$$

From Proposition 2 we have $SIG_A^{out}(a)\ge 0$ and $SIG_A^{in}(a)\ge 0$. Thus, $SIG_A^{out}(a)$ and $SIG_A^{in}(a)$ calculated by changing number of generalized discernibility function when adding attribute set a into A or rejecting set a out of A and $SIG_A^{out}(a)$, $SIG_A^{in}(a)$ is larger the greater amount of changing, or attribute set a is more important and reversing.

To continue, the author recomment heuristic algorithms based on the form of attribute set's level of importance in order to find the best reduct. The ideas of the algorithm initials with empty attribute set $R:=\{\varnothing\}$, repeat adding the most important attribute set into set R until finding reduct.

The algorithm use the strategy Add-Delete [11].

Algorithms 1. Heuristic algorithm find a reduct based on generalized discernibility function.

Input: Set-valued decision systems $DS=(U, AT\cup\{d\})$.

Output: a reduct R.

1. $R=\varnothing$;

//Adding the most important attribute set into set R;

2. While $DIS(R)\ne DIS(AT)$ do

3. Begin
4. For each $a \in AT - R$ calculate

$$SIG_R^{out}(a) = DIS(R \cup \{a\}) - DIS(R);$$

5. Select $a_m \in AT - R$ in order to

$$SIG_R^{out}(a_m) = \underset{a \in AT-R}{Max}\{SIG_R^{out}(a)\};$$

6. $R = R \cup \{a_m\}$;
7. End;
8. For each $a \in R$
9. If $DIS(R - \{a\}) = DIS(R)$ then $R = R - \{a\}$;
10. Return R;

Assume that k is conditional attribute and n is object. The time complexity to calculate M_{AT} is $O(kn^2)$, so the time complexity to calculate $DIS(AT)$ is $O(kn^2)$.
Let us consider While loop from line 2 to line 7, the time complexity to calculate $SIG_R(a)$ is $(k+(k-1)+...+1)*kn^2 = (k*(k-1)/2)*kn^2 = O(k^3n^2)$. The time complexity to selectet the most important attribute is $k+(k-1)+...+1 = k*(k-1)/2 = O(k^2)$. Thus, the time complexity of While loop is $O(k^3n^2)$. Similarity, the time complexity of For loop is $O(k^2n^2)$. Therefore, the time complexity of Algorithm 1 is $O(k^3n^2)$.

Example 5. Let us consider set-valued decision system $DS = (U, AT \cup \{d\})$ in the Example 1. Applying Algorithm 1 to find reduct R, we have:

Set $R = \varnothing$ and calculating:

$$SIG_{\varnothing}^{out}(a_1) = DIS(\{a_1\}) - DIS(\varnothing) = DIS(\{a_1\}) = 0$$
$$SIG_{\varnothing}^{out}(a_2) = DIS(\{a_2\}) - DIS(\varnothing) = DIS(\{a_2\}) = 0$$
$$SIG_{\varnothing}^{out}(a_3) = DIS(\{a_3\}) - DIS(\varnothing) = DIS(\{a_3\}) = 10$$
$$SIG_{\varnothing}^{out}(a_4) = DIS(\{a_4\}) - DIS(\varnothing) = DIS(\{a_4\}) = 4$$

Selecting a_3 which is the most important attribute set and $R = \{a_3\}$. From the example 4, got: $DIS(AT) = 12$, so $DIS(R) \neq DIS(AT)$. Moving to the second loop and calculating:

$$SIG_{\{a_3\}}^{out}(a_1) = DIS(\{a_1, a_3\}) - DIS(\{a_3\}) = 10 - 10 = 0$$
$$SIG_{\{a_3\}}^{out}(a_2) = DIS(\{a_2, a_3\}) - DIS(\{a_3\}) = 10 - 10 = 0$$
$$SIG_{\{a_3\}}^{out}(a_4) = DIS(\{a_3, a_4\}) - DIS(\{a_3\}) = 12 - 10 = 2$$

Selecting a_4 which is the most important attribute set, and $R = \{a_3, a_4\}$. Looking at $DIS(\{a_3, a_4\}) = DIS(AT) = 12$, moving to For loop, testing set R and got. According to the calculation above, $DIS(\{a_4\}) \neq DIS(AT)$ and $DIS(\{a_3\}) \neq DIS(AT)$. Thus, the algorithm ended and $R = \{a_3, a_4\}$ is "the best" reduct of AT.

IV. Attribute reduction in Set-valued decision system when adding or deleting and attribute set.

In this part, the authors research changing of generalized discernibility matrix and generalized discernibility function in set-valued decision systems with two cases: adding and deleting attribute set.

Proposition 4. Let $DS = (U, AT \cup \{d\})$ be a set-valued decision systems, $A, B \subseteq AT$, $A \cap B = \varnothing$ and $U = \{u_1, ..., u_n\}$. Assume that $M_{A \cup B} = (m_{ij}^{AB})_{n \times n}$ and $M_A = (m_{ij}^{A})_{n \times n}$ is generalized discernibility matrix of DS in attribute set $A \cup B$ and A, corresponsively. Then, elements of $M_{A \cup B} = (m_{ij}^{AB})_{n \times n}$ is calculated based on elements of $M_A = (m_{ij}^{A})_{n \times n}$ as follows:

(1) $m_{ij}^{AB} = 1$ if $m_{ij}^{A} = 1$ or $d(u_j) \notin \partial_A(u_i) \cap \partial_B(u_i)$.

(2) $m_{ij}^{AB} = 0$ if $m_{ij}^{A} = 0$ and $d(u_j) \in \partial_A(u_i) \cap \partial_B(u_i)$.

Example 6. Let us consider set-valued decision system $DS = (U, AT \cup \{d\})$ is achived from table 1 by adding attribute set $\{a_5, a_6\}$ as table 2:

Table 2. Set-valued decision system

U	a_1	a_2	a_3	a_4	a_5	a_6	d
u_1	{1}	{1}	{1}	{0}	{0}	{0}	1
u_2	{0}	{0, 1}	{1}	{0}	{1}	{1}	1
u_3	{0, 1}	{0, 1}	{0}	{1}	{0}	{0}	0
u_4	{1}	{0, 1}	{1}	{1}	{0,1}	{0}	1
u_5	{0, 1}	{0, 1}	{1}	{1}	{0}	{1}	2
u_6	{0}	{1}	{1}	{0, 1}	{0}	{0}	1

Suppose that $A = \{a_1, a_2, a_3, a_4\}$, $B = \{a_5, a_6\}$.

From example 2, generalized discernibility matrix of DS in A is:

$$M_A = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & \underline{0} & 0 \\ \underline{0} & \underline{0} & 1 & \underline{0} & 0 & \underline{0} \\ 0 & 0 & 1 & 0 & \underline{0} & 0 \end{bmatrix}$$

Calculate $M_{A \cup B}$ based on Proposition 4. Consider u_6, we have $\partial_A(u_6) = \{1, 2\}$, $\partial_B(u_6) = \{0, 1\}$. So $\partial_A(u_6) \cap \partial_B(u_6) = \{1\}$. According to Proposition 4, $m_{63}^{AB} = m_{63}^{A} = 1$, because of $d(u_1) = d(u_2) = d(u_4) = d(u_6) = 1 \in \{1\}$ so $m_{61}^{AB} = m_{62}^{AB} = m_{64}^{AB} = m_{66}^{AB} = 0$, $d(u_5) = 2 \notin \{1\}$ thus $m_{65}^{AB} = 1$.
Similarity, we have:

$$M_{A \cup B} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & \underline{1} & 0 \\ \underline{1} & \underline{1} & 1 & \underline{1} & 0 & \underline{1} \\ 0 & 0 & 1 & 0 & \underline{1} & 0 \end{bmatrix}$$

Elements underlined in generalized discernibility matrix are elements with data changed. The direct calculating method $M_{A\cup B}$ based on definition 3 have the same result.

Proposition 5. Let $DS=(U, AT\cup\{d\})$ be a set-valued decision system, $B\subset A\subseteq AT$ and $U=\{u_1,...,u_n\}$. Assuming $M_{A-B}=(m_{ij}^{A-B})_{n\times n}$ and $M_A=(m_{ij}^{A})_{n\times n}$ corresponsive is generalized discernibility matrix of DS in attribute set $A-B$ and A. While, elements of $M_{A-B}=(m_{ij}^{A-B})_{n\times n}$ are calculated based on elements of $M_A=(m_{ij}^{A})_{n\times n}$ as followed:

(1) $m_{ij}^{A-B}=1$ if $m_{ij}^{A}=1$ and $d(u_j)\notin\partial_{A-B}(u_i)$.

(2) $m_{ij}^{A-B}=0$ if $m_{ij}^{A}=0$ or $d(u_j)\in\partial_{A-B}(u_i)$.

Example 7. Considering set-valued decision system $DS=(U, AT\cup\{d\})$ in example 6, setting $A=\{a_1,a_2,a_3,a_4,a_5,a_6\}$, $B=\{a_3,a_5\}$.

According to example 6, got:

$$M_A=\begin{bmatrix}0&0&1&0&1&0\\0&0&1&0&1&0\\1&1&0&1&1&1\\0&0&1&0&1&0\\1&1&1&1&0&1\\0&0&1&0&1&0\end{bmatrix}$$

Building M_{A-B} based on Proposition 5. Considering u_6 got $\partial_{A-B}(u_6)=\{0,1\}$. According Proposition 5, $m_{61}^{A-B}=m_{62}^{A-B}=m_{64}^{A-B}=0$, because $d(u_3)=0\in\{0,1\}$ so $m_{63}^{A-B}=0$, $d(u_5)=2\notin\{0,1\}$ thus $m_{65}^{A-B}=1$. Similarity calculating, got:

$$M_{A-B}=\begin{bmatrix}0&0&1&0&1&0\\0&0&1&0&1&0\\\underline{0}&\underline{0}&0&\underline{0}&1&\underline{0}\\0&0&\underline{0}&0&1&0\\1&1&1&1&0&1\\0&0&\underline{0}&0&1&0\end{bmatrix}$$

A direct calculating method $M_{A\cup B}$ based on definition 3 got the same result.

Algorithm 2. Extended algorithm of set-valued decision table when adding attribute set to find reduct.

Input: Set-valued decision system $DS=(U, AT\cup\{d\})$, the best reduct R_{AT} of attribute set AT and attribute set P with $P\cap AT=\varnothing$.

Output: The best reduct $R_{AT\cup P}$ of attribute set $AT\cup P$.

1. $R=R_{AT}$;
2. Calculate $M_{AT\cup P}$ based on Proposition 4; Calculate $DIS(AT\cup P)$;
3. While $DIS(R)\neq DIS(AT\cup P)$ do
4. Begin
   For each $a\in P-R$ calculate $SIG_R^{out}(a)=DIS(R\cup\{a\})-DIS(R)$ with $DIS(R\cup\{a\})$ based on Proposition 4;
5. Select $a_m\in P-R$ in order to $SIG_R^{out}(a_m)=\underset{a\in P-R}{Max}\{SIG_R^{out}(a)\}$;
6. $R=R\cup\{a_m\}$;
7. End;
8. For each $a\in R$
9. If $DIS(R-\{a\})=DIS(AT\cup P)$ then $R=R-\{a\}$;
10. Return R;

Assume that p is an attribute of P and n is an object. According to equation of generalized discernibility matrix in Proposition 4, the time complexity to calculate $M_{R\cup\{a\}}$ if know M_R is $O(n^2)$. So, the time complexity to calculate $DIS(R\cup\{a\})$ if know $DIS(R)$ is $O(n^2)$.

Considering While loop from line 3 to line 7, the time complexity to calculate all $SIG_R^{out}(a)$ is $(p+(p-1)+...+1)*n^2=(p*(p-1)/2)*n^2=O(p^2n^2)$. The time complexity to select the most important attribute is $p+(p-1)+...+1=p*(p-1)/2=O(p^2)$. Thus, the time complexity of While loop is $O(k^3n^2)$. Similarity, the time complexity of For loop is $O(pn^2)$. Therefore, the time complexity of Algorithm 1 is $O(p^2n^2)$. If finding reduct of attribute set $AT\cup P$ by Algorithm 1, then the time complexity of Algorithm 1 is $O((k+p)^3n^2)$. Thus, Algorithm 2 for finding reduct based on extended methods save time to calculate.

Example 8. From example 5, got $\{a_3,a_4\}$ is the best reduct of set-valued decision table in example 1. Considering set-valued decision table $DS=(U, AT\cup\{d\})$ in example 6 (Table 2) with $AT=\{a_1,a_2,a_3,a_4,a_5,a_6\}$, by appling Algorithm 2 to find the best reduct when adding attribute set $P=\{a_5,a_6\}$, got:

Setting $R=\{a_3,a_4\}$, from example 6, calculate $DIS(AT)=18$,

$SIG_{\{a_3,a_4\}}^{out}(a_5)=DIS(\{a_3,a_4,a_5\})-DIS(\{a_3,a_4\})=12-12=0$

$SIG_{\{a_3,a_4\}}^{out}(a_6)=DIS(\{a_3,a_4,a_6\})-DIS(\{a_3,a_4\})=18-12=6$

Selecting a_6 which is the most important attribute set and $R=\{a_3,a_4,a_6\}$. From $DIS(\{a_3,a_4,a_6\})=18$ got $DIS(\{a_3,a_4,a_6\})=DIS(AT)$. Moving *For* loop to test set R, achieves.

Calculate $DIS(\{a_4,a_6\})=12$, so $DIS(\{a_4,a_6\})\neq DIS(AT)$.

Calculate $DIS(\{a_3,a_6\})=13$, so $DIS(\{a_3,a_6\})\neq DIS(AT)$.

Based on the calculation above, $DIS(\{a_3,a_4\})=12$, so $DIS(\{a_3,a_4\})\neq DIS(AT)$.

Thus, algorithm finishes and $R=\{a_3,a_4,a_6\}$ is "the best" reduct of AT.

Algorithm 3. The algorithm to find reduct of set-valued decision table when deleting attribute set.

Input: Set-valued decision system $DS=(U,AT\cup\{d\})$, the best reduct R_{AT} of attribute set AT and attribute set P with $P\subset AT$.

Output: The best reduct R_{AT-P} of attribute set $AT-P$.

1. $R=R_{AT}-P$;
2. Calculate M_{AT-P} followed Proposition 5; Calculate $DIS(AT-P)$;
3. While $DIS(R)\neq DIS(AT-P)$ do
4. Begin
   For each $a\in R$ calculate
   $SIG_R^{in}(a)=DIS(R)-DIS(R-\{a\})$ with
   $DIS(R-\{a\})$ calculate followed Proposition 5;
5. Select $a_m\in R$ in order to
   $SIG_R^{in}(a_m)=\underset{a\in R}{Min}\{SIG_R^{in}(a)\}$;
6. $R=R-\{a_m\}$;
7. End;
8. For each $a\in R$
9. If $DIS(R-\{a\})=DIS(AT-P)$ then $R=R-\{a\}$;
10. Return R;

Similar to Algorithm 2, the time complexity of Algorithm 3 is $O(|R_{AT}-P|^2 n^2)$ where $|R_{AT}-P|$ is cardinality of $R_{AT}-P$.

Example 9. Let us consider set-valued decision table $DS=(U,AT\cup\{d\})$ in Example 6 (Table 2) where $R=\{a_3,a_4,a_6\}$ is the best reduct in Example 8. Appling Algorithm 3 to find the best reduct when deteting attribute set $P=\{a_3,a_5\}$, we have:

Let $R=\{a_3,a_4,a_6\}-\{a_3,a_5\}=\{a_4,a_6\}$, from Example 7, we have $DIS(AT-P)=12$. From Example 8, we have $DIS(\{a_4,a_6\})=12$, so $DIS(\{a_4,a_6\})=DIS(AT-P)$. Go to For loop, we have:

Calculate $DIS(\{a_6\})=6$, so $DIS(\{a_6\})\neq DIS(AT-P)$.

Calculate $DIS(\{a_4\})=4$, so $DIS(\{a_4\})\neq DIS(AT-P)$.

Thus, the algorithm finishes and $R=\{a_4,a_6\}$ is "the best" reduct of $AT-P$.

V. Relation between reduct' definitions

In this part, the authors research the relation between reduct' definition of set-valued decision system. In order to briefing, the author used reduct symbols as following:

R_P - Reduct based on positive region
R_∂ - Reduct based on expansive decision function
R_M - Reduct based on discernibility matrix
R_{DF} - Reduct based on generalized discernibility function
R_{CF} - Reduct based on discernibility function

A. Relation between R_∂ and R_P

Proposition 6. For set-valued decision system $DS=(U,AT\cup\{d\})$ and $R\subseteq AT$. If $\partial_R(u)=\partial_{AT}(u)$ with all $u\in U$ then $POS_R(\{d\})=POS_{AT}(\{d\})$.

Proof: Assuming $POS_R(\{d\})\neq POS_{AT}(\{d\})$, when surely exsiting $u_0\in U$ in order to $u_0\in POS_{AT}(\{d\})$ and $u_0\notin POS_R(\{d\})$. From $u_0\in POS_{AT}(\{d\})$ infering $|\partial_{AT}(u_0)|=1$, from $u_0\notin POS_R(\{d\})$ infering $|\partial_R(u_0)|\neq 1$. Thus, $|\partial_R(u_0)|\neq|\partial_{AT}(u_0)|$. Because of $T_{AT}(u_0)\subseteq T_R(u_0)$ that $\partial_{AT}(u_0)\subseteq\partial_R(u_0)$, combining with $|\partial_R(u_0)|\neq|\partial_{AT}(u_0)|$ infering $\partial_{AT}(u_0)\neq\partial_R(u_0)$. This is contractive with a condition $\partial_R(u)=\partial_{AT}(u)$ with all $u\in U$. So, assuming it is wrong and conclusing if $\partial_R(u)=\partial_{AT}(u)$ with all $u\in U$ then $POS_R(\{d\})=POS_{AT}(\{d\})$.

Note: If DS is inconsistent then another way of Proposition 6 is not satisfied. It is illustrated by example 9.

Example 9. Considering set-valued decision system in table 3

Table 3. Set- valued decision system in example 9

U	a_1	a_2	a_3	*d*
u_1	1	1	0	0
u_2	1	1	0	1
u_3	1	{1, 2}	0	0
u_4	{1, 2}	2	2	2
u_5	2	2	{0, 2}	0

Getting $\partial_{AT}(u_1)=\partial_{AT}(u_2)=\partial_{AT}(u_3)=\{0,1\}$, $\partial_{AT}(u_4)=\partial_{AT}(u_5)=\{0,2\}$. So DS is inconsistent and it is easy to see $POS_{AT}(\{d\})=\varnothing$. Considering $R=\{a_1,a_2\}$, it is easy to see $POS_R(\{d\})=POS_{AT}(\{d\})=\varnothing$.

However, $\partial_{AT}(u_3)=\{0,1\}$ and $\partial_R(u_3)=\{0,1,2\}$, thus $\partial_R(u_3)\neq\partial_{AT}(u_3)$.

Proposition 6 shows that if R_∂ is a reduct based on expansive decision system then existing $R_P\subseteq R_\partial$ with R_P is a reduct in positive region.

If DS is consistent then $POS_R(\{d\}) = POS_{AT}(\{d\}) = U$, means that with all $u \in U$ having $|\partial_R(u)| = |\partial_{AT}(u)| = 1$, or $\partial_R(u) = \partial_{AT}(u)$. So, $\partial_R(u) = \partial_{AT}(u)$ with all $u \in U$ if and only if $POS_R(\{d\}) = POS_{AT}(\{d\})$, this is mean that R_∂ is equivalent with R_P.

B. Relation between R_M and R_∂

In order to research relation between R_M and R_∂, firstly the author demonstrated lemma as below:

Lemma 1. For set-valued decision system $DS = (U, AT \cup \{d\})$, $R \subseteq AT$ and $M_{DS} = [m_{ij}]_{n\times n}$ is discernibility matrix of DS. When this is a condition $R \cap m_{ij} \neq \varnothing$ with $\forall m_{ij} \neq \varnothing$ if and only if $T_R(u_i) - T_{R\cup\{d\}}(u_i) = T_{AT}(u_i) - T_{AT\cup\{d\}}(u_i)$ with $\forall u_i \in U$.

Proof: Considering set-valued decision system $DS = (U, AT \cup \{d\})$ with $U = \{u_1, u_2, ..., u_n\}$ and $R \subseteq AT$.

1) Demonstrating if $T_R(u_i) - T_{R\cup\{d\}}(u_i) = T_{AT}(u_i) - T_{AT\cup\{d\}}(u_i)$ with $\forall u_i \in U$ then $R \cap m_{ij} \neq \varnothing$ with all $m_{ij} \neq \varnothing$.

Assuming to exist $m_{i_0 j_0} \neq \varnothing$ in order to $R \cap m_{i_0 j_0} = \varnothing$. When existing u_{i_0}, $u_{j_0} \in U$ in order to $d(u_{i_0}) \neq d(u_{j_0})$. u_{i_0}, u_{j_0} do not distinguish with each other because of attributes in *R* and u_{i_0}, u_{j_0} distinguish with each other because of attribute in $AT - R$, means that $u_{j_0} \notin T_{AT}(u_{i_0})$ and $u_{j_0} \in T_R(u_{i_0})$.

From $u_{j_0} \notin T_{AT}(u_{i_0})$ infering $u_{j_0} \notin T_{AT}(u_{i_0}) - T_{AT\cup\{d\}}(u_{i_0})$ (*).

From $u_{j_0} \in T_R(u_{i_0})$ và $d(u_{i_0}) \neq d(u_{j_0})$ infering

$u_{j_0} \in T_R(u_{i_0}) - T_R(u_{i_0}) \cap T_{\{d\}}(u_{i_0})$ or

$u_{j_0} \in T_R(u_{i_0}) - T_{R\cup\{d\}}(u_{i_0})$. (**).

From (*) and (**) infering

$T_R(u_i) - T_{R\cup\{d\}}(u_i) \neq T_{AT}(u_i) - T_{AT\cup\{d\}}(u_i)$, this is contractive with the assumption. So that, assumption is wrong and getting a demonstration.

2) Otherwise, the author need to demonstrate if $R \cap m_{ij} \neq \varnothing$ with all $m_{ij} \neq \varnothing$ then $T_R(u_i) - T_{R\cup\{d\}}(u_i) = T_{AT}(u_i) - T_{AT\cup\{d\}}(u_i)$ với $\forall u_i \in U$.

Assuming to exist u_{i_0} in order to $T_R(u_{i_0}) - T_{R\cup\{d\}}(u_{i_0}) \neq T_{AT}(u_{i_0}) - T_{AT\cup\{d\}}(u_{i_0})$.

Because of $T_{AT}(u_{i_0}) - T_{AT\cup\{d\}}(u_{i_0}) \subseteq T_R(u_{i_0}) - T_{R\cup\{d\}}(u_{i_0})$ so existing $u_{j_0} \in U$ in order to $u_{j_0} \in T_R(u_{i_0}) - T_{R\cup\{d\}}(u_{i_0})$ and $u_{j_0} \notin T_{AT}(u_{i_0}) - T_{AT\cup\{d\}}(u_{i_0})$.

From $u_{j_0} \in T_R(u_{i_0}) - T_{R\cup\{d\}}(u_{i_0})$ infering $d(u_{j_0}) \neq d(u_{i_0})$, combining with $u_{j_0} \notin T_{AT}(u_{i_0}) - T_{AT\cup\{d\}}(u_{i_0})$ infering $u_{j_0} \notin T_{AT}(u_{i_0})$. According to discernibility matrix theory, existing $m_{i_0 j_0} \neq \varnothing$ in order to with all $a \in m_{i_0 j_0}$ then $a \notin R$ (because of $u_{j_0} \in T_R(u_{i_0})$), means that $R \cap m_{i_0 j_0} = \varnothing$. This is contractive with the condition $R \cap m_{ij} \neq \varnothing$ with all $m_{ij} \neq \varnothing$. Thus, an assumption is false and getting a demonstration.

From 1) and 2) conclused that $R \cap m_{ij} \neq \varnothing$ with $\forall m_{ij} \neq \varnothing$ if and only if $T_R(u_i) - T_{R\cup\{d\}}(u_i) = T_{AT}(u_i) - T_{AT\cup\{d\}}(u_i)$ with $\forall u_i \in U$.

Proposition 7. For set-valued decision system $DS = (U, AT \cup \{d\})$ and $R \subseteq AT$. If $R \cap m_{ij} \neq \varnothing$ with $\forall m_{ij} \neq \varnothing$ then $\forall u \in U,\ \partial_R(u) = \partial_A(u)$.

Proof: Considering set-valued decision system $DS = (U, AT \cup \{d\})$ with $U = \{u_1, u_2, ..., u_n\}$ and $R \subseteq AT$. According to lemma 1, condition $R \cap m_{ij} \neq \varnothing$ with $\forall m_{ij} \neq \varnothing$ is equivalent with:

$$T_R(u_i) - T_{R\cup\{d\}}(u_i) = T_{AT}(u_i) - T_{AT\cup\{d\}}(u_i) \text{ with } \forall u_i \in U \qquad \textbf{(1)}$$

On the other hand,

$$T_R(u_i) = \left(T_R(u_i) \cap T_{\{d\}}(u_i)\right) \cup \left(T_R(u_i) - \left(T_R(u_i) \cap T_{\{d\}}(u_i)\right)\right)$$

$$T_{AT}(u_i) = \left(T_{AT}(u_i) \cap T_{\{d\}}(u_i)\right) \cup \left(T_{AT}(u_i) - \left(T_{AT}(u_i) \cap T_{\{d\}}(u_i)\right)\right)$$

Assuming $d_i = d(u_i)$,

$$R_i = \left\{d(v_i) \mid v_i \in T_R(u_i) - \left(T_R(u_i) \cap T_{\{d\}}(u_i)\right)\right\},$$

$$AT_i = \left\{d(v_i) \mid v_i \in T_{AT}(u_i) - \left(T_{AT}(u_i) \cap T_{\{d\}}(u_i)\right)\right\}$$

Getting:

$$\partial_R(u_i) = \left\{ \begin{array}{l} d(v_i) \mid v_i \in \left(T_R(u_i) \cap T_{\{d\}}(u_i)\right) \cup \\ \left(T_R(u_i) - \left(T_R(u_i) \cap T_{\{d\}}(u_i)\right)\right) \end{array} \right\} = \{d_i\} \cup R_i$$

$$\partial_{AT}(u_i) = \left\{ \begin{array}{l} d(v_i) \mid v_i \in \left(T_{AT}(u_i) \cap T_{\{d\}}(u_i)\right) \cup \\ \left(T_{AT}(u_i) - \left(T_{AT}(u_i) \cap T_{\{d\}}(u_i)\right)\right) \end{array} \right\} = \{d_i\} \cup AT_i$$

From formula (1) infering $R_i = AT_i$, so $\partial_R(u_i) = \partial_{AT}(u_i)$ with all $u_i \in U$.

Note: If DS is inconsistent then oposite site of Proposition 6 is not satisfied because of the condition $\forall u_i \in U, \partial_R(u_i) = \partial_{AT}(u_i)$ only keep a general decision formular $\partial_R(u_i)$ of tolerance classification $T_R(u_i)$, condition $R \cap m_{ij} \neq \varnothing$ with $\forall m_{ij} \neq \varnothing$ (or $T_R(u_i) - T_{R\cup\{d\}}(u_i) = T_{AT}(u_i) - T_{AT\cup\{d\}}(u_i)$ with $\forall u_i \in U$) keep inconsistent objects with u_i of tolerance classification

$T_R(u_i)$ (strict condition). That is illustrated by example 10 below.

Example 10. Considering set-valued decision system in table 4.

Table 4. Set valued decision system in example 10

U	a_1	a_2	a_3	d
u_1	1	{1, 2}	0	1
u_2	1	1	0	0
u_3	1	{1, 2}	0	0
u_4	{1, 2}	2	2	0
u_5	2	2	{0, 2}	1

Getting: $T_{AT}(u_1)=T_{AT}(u_2)=T_{AT}(u_3)=\{u_1,u_2,u_3\}$,
$T_{AT}(u_4)=T_{AT}(u_5)=\{u_4,u_5\}$,
$\partial_{AT}(u_1)=\partial_{AT}(u_2)=\partial_{AT}(u_3)=\partial_{AT}(u_4)=\partial_{AT}(u_5)=\{0,1\}$
So, DS is inconsistent. Considering $R=\{a_1,a_2\}$ got
$T_R(u_1)=T_R(u_3)=\{u_1,u_2,u_3,u_4\}$, $T_R(u_2)=\{u_1,u_2,u_3\}$,
$T_R(u_4)=\{u_1,u_3,u_4,u_5\}$, $T_R(u_5)=\{u_4,u_5\}$,
$\partial_R(u_1)=\partial_R(u_2)=\partial_R(u_3)=\partial_R(u_4)=\partial_R(u_5)=\{0,1\}$
Thus, with $\forall u_i \in U, i=1..5$, $\partial_R(u_i)=\partial_{AT}(u_i)$. On the other hand, discernibility matrix of DS is:

$$M_{DS}=\begin{bmatrix} 0 & 0 & 0 & \{a_3\} & 0 \\ 0 & 0 & 0 & 0 & \{a_1,a_2\} \\ 0 & 0 & 0 & 0 & \{a_1\} \\ \{a_3\} & 0 & 0 & 0 & 0 \\ 0 & \{a_1,a_2\} & \{a_1\} & 0 & 0 \end{bmatrix}$$

Clearly, $\{a_1,a_2\}\cap\{a_3\}=\varnothing$.

According to Proposition 7 if R_M is a reduct based on discernibility matrix then existing $R_\partial \subseteq R_M$ with R_∂ is a reduct based on expansive decision function.

If DS is consistent, from condition
$\forall u_i \in U, |\partial_R(u_i)|=|\partial_{AT}(u_i)|=1$ infering
$T_R(u_i)=T_{R\cup\{d\}}(u_i)$ and $T_A(u_i)=T_{A\cup\{d\}}(u_i)$ with $\forall u_i \in U$,
so $T_R(u_i)-T_{R\cup\{d\}}(u_i)=T_{AT}(u_i)-T_{AT\cup\{d\}}(u_i)$ with $\forall u_i \in U$.
According to lemma 1, getting $R\cap m_{ij}\neq\varnothing$ with $\forall m_{ij}\neq\varnothing$.
There for, $R\cap m_{ij}\neq\varnothing$ với $\forall m_{ij}\neq\varnothing$ if and only if
$\forall u_i \in U, \partial_R(u_i)=\partial_{AT}(u_i)$, means is R_M is equivalent with R_∂.

VI. Conclusion

In this paper, based on discernibility matrix and discernibility function in traditional rough set theory [8], the authors proposed generalized discernibility matrix and generalized discernibility function in order to find reduct of static set-valued decision system.

We also proposed increasing algorithms with dynamically-increasing and decreasing conditional attributes. Increasing method reduces significantly the execution time of finding reduct, caused of avoiding repetition it in the whole set-attributions. Further research is building increasing algorisms with dynamically-increasing or decreasing set-object in order to find rough set in static set-valued decision systems, the author researched relation between reduct' definitions in set-valued decision system.

References

[1] Chen Z. C, Shi P., Liu P. G., Pei Z., Criteria Reduction of Set-Valued Ordered Decision System Based on Approximation Quantity, *International Journal of Innovative Computing, Information and Control*, Vol 9, N 6, pp. 2393-24-4, 2013.

[2] Guan Y. Y., Wang H. K., Set-valued information systems, *Information Sciences 176*, pp. 2507–2525, 2006.

[3] Kryszkiewicz M., Rough set approach to incomplete information systems, *Information Science*, Vol. 112, pp. 39-49, 1998.

[4] Pawlak Z., Rough sets, *International Journal of Information and Computer Sciences*, 11(5), pp. 341-356, 1982.

[5] Pawlak Z., Rough sets: Theoretical Aspects of Reasoning About Data, *Kluwer Academic Publishers*, 1991.

[6] Qian Y. H., Dang C. Y., Liang J. Y., Tang D. W., Set-valued ordered information systems, *Information Sciences 179,* pp. 2809-2832, 2009.

[7] Sinh Hoa Nguyen and Thi Thu Hien Phung., Efficient Algorithms for Attribute Reduction on Set-valued Decision Systems, The 14th International Conference of Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC 2013), Halifax, NS, Canada, October 11-14, 2013, *Lecture Notes of Computer Science, SpingerLink*, Volume 8170 , pp 87-98, 2013.

[8] Shifei D., Hao D., Research and Development of Attribute Reduction Algorithm Based on Rough Set, *IEEE,* CCDC2010, pp.648-653, 2010.

[9] Skowron A., Rauszer C., The Discernibility Matrices and Functions in Information systems, *Interlligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht,* pp. 331-362, 1992.

[10] Y. H. Qian Y. H. , Liang J. Y., On Dominance Relations in Disjunctive Set-Valued Ordered InformationSystems, *International Journal of Information Technology & Decision Making* Vol. 9, No. 1, pp. 9–33, 2010.

[11] Yao Y.Y., Zhao Y., Wang J., On reduct construction algorithms, *Proceedings of International Conference on Rough Sets and Knowledge Technology*, pp. 297-304, 2006.

[12] Zhang J. B., Li T. R., Ruan D., Liu D., Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems, *International Journal of Approximate Reasoning 53*, pp. 620–635, 2012.

Author Biographies

Thi Thu Hien Phung was born in Viet Nam on November 30, 1972. She is a lecturer in the Department of Information Technology at the University of Economic and Technical Industries, VietNam. Her area of research is Rough Set theory.

Van Tho Ninh was born in Viet Nam on April 02, 1972. He is a lecturer in the Department of Electrical Electronics Technology at the University of Economic and Technical Industries, VietNam.