Relation between definition of Reduct in Set-valued Decision System

Thi Thu Hien Phung¹, Thanh Dai Tran²

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

Abstract: In term of researching attribute reduction based on tolerance rough set, this paper researches relation between definitions of reduct in set-valued decision system. The results help us deeply understand characteristic of reduct’s definition in set-valued decision system, also it is a theory foundation in order to create new and efficiency methods.

For attribute reduction problems, the most important issue is to minimize the time complexity of attribute reduction algorithms. There have been many methods for attribute reduction, but finding reducts in these methods is almost performed on initial object set. In this paper, we propose a method for selection representative object set from initial object set to solve attribute reduction problem in set-valued decision system.

Keywords: Tolerance rough set, decision system, attribute reduct, reduct.

I. Introduction

Pawlak’s rough set theory [5] suggests unreliable definition instrument and broadly apply for researching data.

In real information systems, attribute value of object might be a valued set. For example, considering an object in the information system ‘Nguyen Van A’ at attribute ‘Language’ is ‘English, French, Russian’, means that Nguyen Van A know language English, or French, or Russian. The information system like that is called set valued information system. In set-valued information system, Yan Yong Guan and co-worker [1] broaden equivalent relation in traditional rough set into tolerance and create tolerance set model.

According to rough set model, some results of reduct’s method in set-valued decision system presented [1, 6, 7, 8, 11]. Each method also gives reduct’s definition and creates heuristic mathematics to find the best reduct which uses attribute’s quality classification standard (or called level of attribute important).

There are some symbolic definition of reduct in set-valued decision system based on rough set model: reduct based on positive region created by positive region’s definition in traditional rough set theory [5], reduct based on generalized decision function which uses expansive decision function [2]. According to discernibility matrix and discernibility function in traditional rough set theory [9], authors in [14] show a definition of reduct in set-valued decision system based on discernibility matrix. In the research [6], the author creates discernibility matrix and discernibility function, after that giving a definition of reduct based on discernibility function.

In the research [8], authors build contingency table and discernibility function in set-valued decision system, after that building a definition of reduct which used discernibility function.

Reduct is a result of a reduct’s method. Thus, it is important to find the relation between reducts which helps to compare and evaluate attribute reduction. However, in set-valued decision system, its research’s result is limited.

Some results are: the authors in [6] proved that reduct which uses expansive discernibility function and generalized decision function is similar. The authors in [8] showed that reduct which used discernibility function and reduct based on discernibility matrix is similar.

Until now, there are many methods to calculate for singer-valued information system [10], but all of them are based on the first objective set. In set-valued information system, the researches [2, 13] presented solutions data squeeze in order to minimize size of the first objective set and reducing time of attribute reduction algorithms. However, a limitation of data squeeze method is no demonstrating attribution reduction in the first objective set which is similar with reduction set after squeeze.

According to the ideas of minimizing the first set’s size [3], in this paper we propose methods of selecting representative object set for attribute reduction in set-valued decision system. We demonstrate reduction in the first object set and representative object set which is similar in set-valued decision system. Because of object set’s size is smaller than the first object set, so reducing the time complexity of attribute reduction algorithms in representative object set. Representative object set’s size selected is depended on characteristic of each information system in real works.

In this paper, section 2 shows basic definition of rough set model in set-valued information system and reduct’s definitions. Section 3 presents the relation between reduct’s
definitions, separating and evaluating methods of attribute reduction. Section 4 proposes methods of selecting representative object set in set-valued decision system. Section 5 is a conclusion and further research.

II. Basic Definitions

In this section, we outline basic concepts of set-valued information systems [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, i.e. the function of form \( a: U \rightarrow V_a \), where \( a(x) \in V_a \) for each \( a \in A \). The set \( V_a \) is called the domain of attribute \( a \in A \). Such information systems will be called the single-valued information systems.

The set-valued information systems have been proposed as a tool to characterize uncertain information or missing information. The idea is to change the definition of attribute, i.e., \( a: U \rightarrow 2^V \). In other words \( a(x) \subseteq V_a \) for any \( x \in U \) and \( a \in A \).

There are many ways to give a semantic interpretation of the set-valued information systems. The most useful interpretations, which are often discussed in literature [12] are:

Conjunctive interpretation: For \( x \in U \) and \( a \in A \), \( a(x) \) is interpreted conjunctively. For example, if \( a \) is the attribute "familiarizing program languages", then the attribute value \( a(u) = [C++, Java, Pascal] \) can be interpreted as \( u \) knows three program languages: C++, Java and Pascal.

Disjunctive interpretation: For \( x \in U \) and \( a \in A \), \( a(x) \) is interpreted disjunctively. For the example above, the value set \( a(x) = [C++, Java, Pascal] \) can be interpreted as \( u \) knows only one of languages C++, Java or Pascal. The value of a numeric attribute "ages" \( b(x) = [20, 25] \) can be interpreted as the person \( u \) has an age between 20 and 25. Incomplete information systems with some unknown attribute values or partial known attribute values are such type of set-valued information systems.

Hybrid interpretation: Combining of the above two models. Some attributes in the information system are interpreted conjunctively, for example an attribute "familiarizing program languages" and some other ones are interpreted disjunctively such as a numeric attribute "ages".

This part presents some basic definitions of tolerance rough set model in set-valued information system [1] and some definitions of reduct in set-valued decision system.

Information system is an tuple \( IS = (U, AT) \), where \( U \) is a finite set of objects and \( AT \) is finite set of attribute. The value of an attribute \( a \in AT \) at an objects \( u \in U \) is denoted as \( a(u) \). 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 is denoted as \([u]_A \), while

\[
[u]_A = \{v \in U \mid (u,v) \in IND(A)\}.
\]

It is easy to see that \([u]_A = \cap [u]_{A|u}\) with all equivalence class in the partition \( U / A \) which includes \( u \in U \ a \in A \).

Considering information system \( IS = (U, AT) \), if existing \( u \in U \) in order to \( a(u) \) contain at least two values then \( IS = (U, AT) \) is called set-valued information system. Each sub attribute set \( A \subseteq AT \), one binary relationship on \( U \) is defined as following below:

\[
T_A = \{(u,v) \in U \times U \mid \forall a \in A, a(u) \cap a(v) \neq \emptyset\}.
\]

It is easy to see that \( T_A \) is not an equivalent relation because of being reflected, symmetric, no transitive attitude. \( T_A \) is called tolerance relationship and \( T_A = \bigcap_{a \in A} T_a \).

Setting \( T_A(u) = \{v \in U \mid (u,v) \in T_A\}, T_A(u) \) is called a classification of tolerance. The symbol \( U / T_A = \{T_A(u) \mid u \in U\} \) presents all classification of tolerance which created by relationship \( T_A \), when \( U / T_A \) creates a site of \( U \) because of classification in \( U / T_A \) can be mix and \( \bigcup_{u \in u \in U} T_A(u) = U \). It is easy too see if \( B \subseteq A \) then \( T_A(u) \subseteq T_B(u) \) with all \( u \in U \).

Similar as traditional rough set theory [4], with \( B \subseteq A \), \( X \subseteq U \), \( B \)-lower approximation of \( X \) is set \( BX = \{u \in U \mid T_B(u) \subseteq X\} = \{u \in X \mid T_B(u) \subseteq X\} \).

\( B \)-upper approximation of \( X \) is set \( \overline{BX} = \{u \in U \mid T_B(u) \cap X \neq \emptyset\} = \bigcup_{u \in U} T_B(u) \cap \{u \in U\} \), \( B \)-Border region of \( X \) is set \( BN_B(X) = \overline{BX} - BX \).

With all approximate set, calling \( B \)-positive region for \( \{d\} \) is set: \( POS_B(\{d\}) = \bigcup_{x \in \{d\}} (BX) \).

Set-valued decision system is set-valued information system \( DS = (U, AT \cup \{d\}) \) inside \( AT \) is conditional attribution and \( d \) is decision attribution, with an assumption \( d(u) \) contains one value with all \( u \in U \). Similar as set uncompleted decision system [2], with \( u \in U \), \( \hat{d}_{\nu}(u) = \{d(v) \mid v \in T_{\nu}(u)\} \) is called expansive decision set of object \( u \) on attribut set \( AT \).

If \( \hat{d}_{\nu}(u) = \{d\} \) with all \( u \in U \) then \( DS \) is consistent, opposite \( DS \) is inconsistent. According to positive region theory, \( DS \) is consistent if and only if \( POS_{\nu}(\{d\}) = U \), opposite \( DS \) is inconsistent.

Reduct is a basic theory in rough set theory. In general, reduct is a smallest sub-set of condition attribute which keep the classic information of decision set. Next, the authors present
some definition about reduct of set-valued decision system which is used in this article.

By using positive region theory in set-valued decision system, the authors give a definition of reduct based on positive region theory.

**Definition 2.1.** For set-valued decision system 

\( DS = (U, AT \cup \{d\}) \). If \( R \subseteq AT \) satisfies:

1. \( POS_R \{d\} = POS_{AT} \{d\} \)
2. \( \forall R \subseteq R, \ POS_R \{d\} \neq POS_{AT} \{d\} \).

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

Using expansive decision function in [2], reduct which based on set valued decision is defined as:

**Definition 2.2.** For set-valued decision system 

\( DS = (U, AT \cup \{d\}) \). If \( R \subseteq AT \) satisfies:

1. \( \partial_R (u) = \partial_{AT} (u) \) with all \( u \in U \).
2. \( \forall R \subseteq R \), existing \( u \in U \) in order to \( \partial_R (u) \neq \partial_{AT} (u) \) then \( R \) is called a reduct of **DS based on expansive decision system**.

According to rough set model in set-valued information system, the authors in [14] give a definition of reduct in discernibility matrix. Discernibility matrix of set-valued decision system \( DS \) is \( M_{DS} = \{ m_{ij} \}_{nn \times} \), elements \( m_{ij} \) determined as:

\[ m_{ij} = \begin{cases} \{ d | a \in AT, (u_i, u_j) \notin T_{AT} \} & d(u_i) \neq d(u_j) \\ \emptyset & \text{otherwise} \end{cases} \]

**Definition 2.6.** For set-valued decision system 

\( DS = (U, AT \cup \{d\}) \) and discernibility matrix 

\( M_{DS} = \{ m_{ij} \}_{nn \times} \). If \( R \subseteq C \) satisfies:

1. \( R \cap m_{ij} \neq \emptyset \) with all \( m_{ij} \neq \emptyset \)
2. With all \( r \in R \), \( R \setminus \{r\} \) do not satisfy (1)

then \( R \) is called as a **reduct of DS based on discernibility matrix**.

Approaching to matrix way, the authors in [8] build discernibility function from contingency table and gave a definition of **reduct based on discernibility**. According to definition of discernibility matrix and discernibility function in traditional rough set theory [9], the author [6] gave a definition of **reduct based on generalized discernibility function**.

**III. Relation between reduct’ definitions**

In this part, the authors evaluate and research the relation between reduct’ definition of set-valued decision system in part 2. In order to briefing, the authors used reduct symbols as following:

<table>
<thead>
<tr>
<th>Table 3.1. Symbol of reduct in set-valued decision system</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Symbol</strong></td>
</tr>
<tr>
<td>( R_p )</td>
</tr>
<tr>
<td>( R_\emptyset )</td>
</tr>
<tr>
<td>( R_M )</td>
</tr>
<tr>
<td>( R_{DF} )</td>
</tr>
<tr>
<td>( R_{CF} )</td>
</tr>
</tbody>
</table>

In set-valued decision system, the author [6] proved \( R_{DF} \) is similar with \( R_\emptyset \), the authors [8] showed \( R_{CF} \) is similar with \( R_M \). Nextly, the authors research a rest of relation between reducts in terms of consistent or inconsistent set valued decision system.

**3.1. Relation between \( R_\emptyset \) and \( R_p \)**

**Clause 3.1.** 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 existing \( u_0 \in U \) in order to \( u_0 \in POS_{AT} \{d\} \) and \( u_0 \in POS_R \{d\} \). From \( u_0 \in POS_{AT} \{d\} \) inferring \( [\partial_{AT} (u_0)] = 1 \), from \( u_0 \in POS_R \{d\} \) inferring \( [\partial_R (u_0)] = 1 \). Thus, \( [\partial_R (u_0)] = [\partial_{AT} (u_0)] \).

Because of \( T_{AT} (u_0) \subseteq T_R (u_0) \) that \( \partial_R (u_0) \subseteq \partial_{AT} (u_0) \), combining with \( [\partial_R (u_0)] = [\partial_{AT} (u_0)] \) inferring \( \partial_R (u_0) \neq \partial_{AT} (u_0) \). This is contractive with a condition \( \partial_R (u) = \partial_{AT} (u) \) with all \( u \in U \). So, assuming it is wrong and concluding 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 clause 3.1 is not satisfied. It is illustrated by example 3.1.

**Example 3.1.** Considering set-valued decision system in table 3.2.

<table>
<thead>
<tr>
<th>Table 3.2. Set-valued decision system in example 3.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>( U )</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>( u_1 )</td>
</tr>
<tr>
<td>( u_2 )</td>
</tr>
<tr>
<td>( u_3 )</td>
</tr>
<tr>
<td>( u_4 )</td>
</tr>
<tr>
<td>( u_5 )</td>
</tr>
</tbody>
</table>

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 \( POS_{AT} \{d\} = \emptyset \). Considering \( R = \{a_1, a_2\} \), it is easy to see \( POS_R \{d\} = POS_{AT} \{d\} = \emptyset \).
However, \( \partial_{AT}(u_3) = \{0,1\} \) and \( \partial_{u}(u_3) = \{0,1,2\} \), thus \( \partial_{u}(u_3) \neq \partial_{AT}(u_3) \).

Clause 3.1 shows that if \( R_0 \) is a reduct based on expansive decision system then existing \( R_P \subseteq R_0 \) with \( R_P \) is a reduct in positive region.

If \( DS \) is consistent then \( \text{POSA}_{ DS}(\{d\}) = \text{POSA}_{ AT}(\{d\}) = U \), means that with all \( u \in U \) having \( |\partial_{u}(u)| = |\partial_{AT}(u)| = 1 \), or \( \partial_{u}(u) = \partial_{AT}(u) \). So, \( \partial_{u}(u) = \partial_{AT}(u) \) with all \( u \in U \) if and only if \( \text{POSA}_{ DS}(\{d\}) = \text{POSA}_{ AT}(\{d\}) \), this is mean that \( R_0 \) is equivalent with \( R_P \).

3.2. Relation between \( R_M \) and \( R_0 \)

In order to research relation between \( R_M \) and \( R_0 \), firstly the authors demonstrated lemma as below:

**Lemma 3.1.** For set-valued decision system \( DS = \{(U, AT \cup \{d\}) \} \), \( R \subseteq AT \) and \( M_M = \{m_{ij}, \alpha\} \) is discernibility matrix of DS. When this is a condition \( R \cap m_i / \neq \emptyset \) with \( \forall m_{ij} \neq \emptyset \) if and only if \( T_R(u) - T_R(\{d\}) = T_AT(u) - T_AT(d) \) 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 \( T_R(u) - T_R(\{d\}) = T_AT(u) - T_AT(d) \) \( \forall u_i \in U \) then \( R \cap m_i \neq \emptyset \) with all \( m_{ij} \neq \emptyset \).

Assuming to exist \( m_{ij} \neq \emptyset \) in order to \( R \cap m_{ij} = \emptyset \) .

When existing \( u_i, u_j \in U \) in order to \( d(u_i) \neq d(u_j) \). \( u_i, u_j \) do not distinguish with each other because of attributes in \( R \) and \( u_i, u_j \) distinguish with each other because of attribute in \( AT-R \), means that \( u_i \neq AT(u_i) \) and \( u_j \neq AT(u_j) \). From \( u_i \neq AT(u_i) \) \( u_j \neq AT(u_j) \) \( u_i \neq AT(u_i) - T_{AT}(d) \) \( u_j \neq AT(u_j) - T_{AT}(d) \) (\( \ast \)).

From \( u_i \neq AT(u_i) \) \( u_j \neq AT(u_j) \) \( d(u_i) \neq d(u_j) \) \( u_i \neq T_R(u_j) \circ T_R(d) \) or \( u_j \neq T_R(u_i) - T_{R\cup\{d\}}(u_i) \).

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

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

Assuming to exist \( u_i \) in order to \( T_R(u_i) - T_{R\cup\{d\}}(u_i) \neq T_AT(u_i) - T_AT(d) \). Because of \( T_AT(u_i) - T_AT(d) \subseteq T_R(u_i) - T_{R\cup\{d\}}(u_i) \) so existing \( u_i \in U \) in order to \( u_i \neq T_R(u_i) - T_{R\cup\{d\}}(u_i) \) and \( u_i \neq T_AT(u_i) - T_AT(d) \).

From \( u_i \neq T_R(u_i) - T_{R\cup\{d\}}(u_i) \) \( d(u_i) \neq d(u_i) \) \( u_i \neq T_AT(u_i) - T_AT(d) \) \( d(u_i) \neq d(u_i) \). According to discernibility matrix theory, existing \( m_{ij} \neq \emptyset \) in order to with all \( a \in m_{ij} \), then \( a \in R \) (because of \( u_i \neq T_R(u_i) \)), means that \( R \cap m_{ij} = \emptyset \). This is contractive with the condition \( R \cap m_{ij} \neq \emptyset \) with all \( m_{ij} \neq \emptyset \). Thus, an assumption is false and getting a demonstration.

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

**Clause 3.2.** For set-valued decision system \( DS = \{(U, AT \cup \{d\}) \} \) and \( R \subseteq AT \). If \( R \cap m_{ij} \neq \emptyset \) with \( \forall m_{ij} \neq \emptyset \) then \( \forall u_i \in U \), \( \partial_R(u_i) = \partial_{AT}(u_i) \).

**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 3.1, condition \( R \cap m_{ij} \neq \emptyset \) is equivalent with:

\( T_R(u_i) - T_{R\cup\{d\}}(u_i) = T_AT(u_i) - T_AT(d) \) \( \forall u_i \in U \).

On the other hand,

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

\[ T_AT(u_i) = \left( T_AT(u_i) \cap T_{d\cup\{d\}}(u_i) \right) \cup \left( T_AT(u_i) - \left( T_AT(u_i) \cap T_{d\cup\{d\}}(u_i) \right) \right) \]

\[ d_i = d(u_i) \cap R_i = \left( d(u_i) \cup R_i \right) \cap \left( T_AT(u_i) \cap T_{d\cup\{d\}}(u_i) \right) \]

\[ AT_i = \left( d(u_i) \cup R_i \right) \cap \left( T_AT(u_i) \cap T_{d\cup\{d\}}(u_i) \right) \]

Getting:

\[ \partial_R(u_i) = \left( d(u_i) \cup R_i \right) \cap \left( T_AT(u_i) \cap T_{d\cup\{d\}}(u_i) \right) \]

\[ \partial_{AT}(u_i) = \left( d(u_i) \cup R_i \right) \cap \left( T_AT(u_i) \cap T_{d\cup\{d\}}(u_i) \right) \]
Relation between definition of Reduct in Set-valued Decision System

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 opposite site of clause 3.2 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 formula $\partial_R(u_i)$ of tolerance classification $T_R(u_i)$, condition $R \cap m_j \neq \emptyset$ with $\forall m_j \neq \emptyset$ (or $T_R(u_i) - T_R(\emptyset(u_i)) = T_{AT}(u_i) - T_{AT}(\emptyset(u_i))$ 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 3.2 below.

**Example 3.2.** Considering set-valued decision system in table 3.3.

<table>
<thead>
<tr>
<th>$U$</th>
<th>$a_1$</th>
<th>$a_2$</th>
<th>$a_3$</th>
<th>$d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>1</td>
<td>${1, 2}$</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$u_3$</td>
<td>1</td>
<td>${1, 2}$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$u_4$</td>
<td>${1, 2}$</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>$u_5$</td>
<td>2</td>
<td>2</td>
<td>${0, 2}$</td>
<td>1</td>
</tr>
</tbody>
</table>

Getting: $T_{AT}(u_1) = T_{AT}(u_2) = T_{AT}(u_3) = \{u_4, u_5\}$, $T_R(u_3) = T_{AT}(u_3) = \{u_4, u_5\}$, $\partial_R(u_3) = \partial_{AT}(u_3) = \partial_{AT}(u_3) = \partial_{AT}(u_3) = \{0, 1\}$.

So, $DS$ is inconsistent. Considering $R = \{a_1, a_3\}$ got $T_P(u_1) = T_P(u_2) = T_P(u_3) = \{u_4, u_5\}$, $T_R(u_3) = \{u_4, u_5\}$, $\partial_R(u_3) = \partial_R(u_3) = \partial_R(u_3) = \partial_R(u_3) = \{0, 1\}$.

Thus, with $\forall u_i \in U, i = 1, 5, \partial_R(u_i) = \partial_R(u_i)$. On the other hand, indiscernibility matrix of $DS$ is:

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

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

According to clause 3.2 if $R_M$ is a reduct based on discernibility matrix then existing $R_3 \subseteq R_M$ with $R_3$ 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 $T_R(u_i) = T_{AT}(u_i)$ and $T_R(u_i) = T_{AT}(u_i)$ with $\forall u_i \in U$, so $T_{AT}(u_i) - T_{AT}(\emptyset(u_i)) = T_{AT}(u_i) - T_{AT}(\emptyset(u_i))$ with $\forall u_i \in U$.

According to lemma 3.1, getting $R \cap m_j \neq \emptyset$ with $\forall m_j \neq \emptyset$. There for, $R \cap m_j \neq \emptyset$ if and only if $\forall u_i \in U, \partial_R(u_i) = \partial_{AT}(u_i)$. means is $R_M$ is equivalent with $R_3$.

### 3.3 Summarizing the relation between reduct definitions of set-valued decision system

Based on the researches, the authors summarize the relation between reduct definitions of set-valued decision system:

If decision system is consistent, reducts $R_P, R_3, R_M, R_{DF}, R_{CF}$ are similar.

If decision table is inconsistent, the relation between presented by a model as followed:

**Model 3.1. Relation between reducts of set-valued decision system**

- **Group 1:** Including reduct $R_P$.
- **Group 2:** Including reducts $R_3, R_{DF}$.
- **Group 3:** Including reducts $R_M, R_{CF}$.

Relation of reducts into groups as:

If $R_3$ is a reduct of group 3 then existing a reduct $R_3$ belong to group 2 and a reduct $R_1$ belong to group 1 in order to $R_1 \subseteq R_2 \subseteq R_3$.

### 3.4 Evaluating methods of attribute reduction in set-valued decision system based on reduct

After giving reduct’s definition, methods of attribute reduction build heuristic algorithm to find the best reduct based on level of important standard of attribute, or quality of attribute classification.

With set-valued consistent decision system, the best reducts of three methods are the same, so they have similar quality of classifications. With set-valued inconsistent decision system, the authors evaluate three methods based on reduct classification’s quality standard.

Assuming $R_{MBest}$ is a reduct of method which belong to Group 3 (found by heuristic algorithms which used discernibility matrix, or discernibility function). Following the results, existing a reduct based on expansive decision function $R_{CF}$ in order to $R_{CF} \subseteq R_{MBest}$ ($R_{CF}$ is minimum than $R_{MBest}$).

Assuming is the best reduct of the second method.
(\(R_{\text{Best}}\) found by heuristic algorithms which used expansive decision function or expansive discernibility function). There are two cases.

- If \(R_{\text{Best}}\) is \(R_{0}\) (\(R_{\text{Best}} = R_{0}\)) then \(R_{\text{Best}} \subseteq R_{\text{MBest}}\), means that \(R_{\text{Best}}\) is minimum than \(R_{\text{MBest}}\) . Mean that , or quality classification of \(R_{\text{Best}}\) is better than \(R_{\text{MBest}}\).

- If \(R_{\text{Best}}\) is different with \(R_{0}\) then quality classification of \(R_{\text{Best}}\) is better than \(R_{0}\) caused of quality classification of \(R_{\text{Best}}\) is the best one. On the other hand, caused of \(R_{0} \subseteq R_{\text{MBest}}\) so \(R_{0}\) is better than \(R_{\text{MBest}}\) about quality classification. Thus, \(R_{\text{Best}}\) is better than \(R_{\text{MBest}}\) about quality classification.

Therefore, in both two cases, quality classification of \(R_{\text{Best}}\) is better than \(R_{\text{MBest}}\). Thus, to summerize, the methods of group 2 is more efficiency than method of group 3 which followed by quality classification of reduct standard.

Similarity, the methods of group 1 is more efficiency than method of group 2 which followed by quality classification of reduct standard. Thus, the method of finding reduct based on generalized discernibility function in [6] is better than discernibility function in [8] which followed by quality classification of reduct standard.

IV. Selecting of representative object set for attribute reduction in set-valued decision system

Selecting a representative object set is essentially data preprocessing step in the data mining and machine learning. Instead of finding a reduct on overall initial object set, we find the reduct based on representative object set and prove theoretically that the reduct obtained from the representative object equals to the reduct obtained from the initial representative object set. Because the size of the representative object is much smaller that of initial object set, the time complexity of attribute reduction algorithm is much smaller. Representative object set consists of representative objects; each object is select as follow:

Let us consider the set-valued decision system \(DS = (U, AT \cup \{d\})\), based on attribute set \(AT\), we first partition the initial set of objects \(U\) into equivalence classes. Two objects \(u, v \in U\) belong to the same equivalence class if \(T_{\{a\}}(u) = T_{\{a\}}(v)\) for all \(a \in AT\). We calculate the partition \(X_{i} / \{d\} = \{Y_{1}, ..., Y_{k}\}\). With equivalence class \(Y_{j} \in X_{i} / \{d\}\), we select a representative object, without generality, we select the first object which is a representative object. Then, selected object set is representative object set.

The algorithm for selecting representative object set in set-valued decision system is presented as follow:

**Algorithm 1.** Selecting representative object set of a set-valued decision system.

**Input:** The initial set-valued decision system \(DS = (U, AT \cup \{d\})\) with \(U = \{u_{1}, ..., u_{n}\}\), \(AT = \{a_{1}, ..., a_{m}\}\).

**Output:** The representative set-valued decision system \(DS_{p} = (U_{p}, AT \cup \{d\})\) with \(U_{p} \subseteq U\) is representative object set.

**Step 1:** Set \(U_{p} = \emptyset\);

**Step 2:** For any \(a_{i} \in AT, i = 1..m\), calculate partition \(U / \{a_{i}\} = \{[u]_{a_{i}} | u \in U\}\) where \(\{u]_{a_{i}} = \{v \in U | T_{\{a\}}(u) = T_{\{a\}}(v)\}\).

**Step 3:** Calculate partition \(U / AT = \{[u]_{AT} | u \in U\}\) where \(\{u]_{AT} = \cap_{i=1}^{m} [u]_{a_{i}}\).

Assume that \(U / AT = \{X_{1}, ..., X_{k}\}\);

**Step 4:** For \(X_{i} \in U / AT, i = 1..k\), do Step 4.1 and 4.2 as follow:

**Step 4.1.** Calculate partition \(X_{i} / \{d\} = \{[u]_{d} | u \in X_{i}\}\) with \(\{u]_{d} = \{v \in X_{i} | d(u) = d(v)\}\).

Assume that \(X_{j} / \{d\} = \{Y_{1}, ..., Y_{j}\}\)

and \(Y_{j} = \{u_{1}, ..., u_{j}\}\) with \(j = 1..I\).

**Step 4.2.** With each \(Y_{j} \in X_{i} / \{d\}\), \(j = 1..I\), set \(U_{p} := U_{p} \cup \{u_{j}\}\);

**Step 5:** Return \(DS_{p} = (U_{p}, AT \cup \{d\})\);

Assume that \(k\) is condition attribute and \(n\) is object. At step 2, for each \(a_{i} \in A, i = 1..m\), the time complexity to calculate \(T_{\{a\}}(u)\) \(u \in U\) is \(O(n^{2})\), the time complexity to calculate \(U / \{a_{i}\}\) is \(O(n\log n)\). Thus, the complexity of Step 2 is \(O(kn^{2})\). The complexity of Step 3 is \(O(n)\). The complexity of Step 4 is \(O(kn\log n)\).

Therefore, the time complexity of algorithm 1 is \(O(kn^{2})\).

**Example 4.1.** Considering set-valued decision system \(DS = (U, AT \cup \{d\})\) in Table 4.1
Table 4.1. Set-valued decision system

<table>
<thead>
<tr>
<th>U</th>
<th>a₁</th>
<th>a₂</th>
<th>a₃</th>
<th>a₄</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td>{1}</td>
<td>{1}</td>
<td>{1}</td>
<td>{0}</td>
<td>1</td>
</tr>
<tr>
<td>u₂</td>
<td>{0}</td>
<td>{0, 1}</td>
<td>{1}</td>
<td>{0}</td>
<td>1</td>
</tr>
<tr>
<td>u₃</td>
<td>{0, 1}</td>
<td>{0, 1}</td>
<td>{0}</td>
<td>{1}</td>
<td>0</td>
</tr>
<tr>
<td>u₄</td>
<td>{1}</td>
<td>{0, 1}</td>
<td>{1}</td>
<td>{1}</td>
<td>1</td>
</tr>
<tr>
<td>u₅</td>
<td>{0, 1}</td>
<td>{0, 1}</td>
<td>{1}</td>
<td>{1}</td>
<td>2</td>
</tr>
<tr>
<td>u₆</td>
<td>{0}</td>
<td>{1}</td>
<td>{1}</td>
<td>{0, 1}</td>
<td>1</td>
</tr>
<tr>
<td>u₇</td>
<td>{0, 1}</td>
<td>{1}</td>
<td>{0}</td>
<td>{0, 1}</td>
<td>0</td>
</tr>
<tr>
<td>u₈</td>
<td>{0}</td>
<td>{1}</td>
<td>{1}</td>
<td>{0}</td>
<td>1</td>
</tr>
<tr>
<td>u₉</td>
<td>{0, 1}</td>
<td>{0, 1}</td>
<td>{0}</td>
<td>{1}</td>
<td>0</td>
</tr>
</tbody>
</table>

We have:

U / AT = \{\{u₁\}, \{u₂, u₆\}, \{u₃, u₉\}, \{u₄\}, \{u₅\}, \{u₆\}, \{u₇\}\}

T\{u₁\}(u₁) = T\{u₁\}(u₂) = \{u₁, u₃, u₄, u₅, u₆, u₇, u₉\},

T\{u₃\}(u₃) = T\{u₅\}(u₅) = T\{u₇\}(u₇) = T\{u₉\}(u₉) = U,

T\{u₁\}(u₆) = T\{u₃\}(u₆) = T\{u₅\}(u₆) = \{u₂, u₃, u₅, u₆, u₇, u₉\}.

Consequently:

U / \{a₁\} = \{\{u₁, u₄\}, \{u₂, u₆, u₈\}, \{u₃, u₅, u₇\}\}.

Similarly, we have U / \{a₂\} = U.

U / \{a₃\} = \{\{u₁, u₂, u₄, u₅, u₆, u₈\}, \{u₃, u₅, u₇\}\},

U / \{a₄\} = \{\{u₁, u₂, u₆\}, \{u₃, u₄, u₅, u₉\}, \{u₆, u₇\}\}.

Then we have:

U / AT = \{\{u₁\}, \{u₂, u₆\}, \{u₃, u₉\}, \{u₄\}, \{u₅\}, \{u₆\}, \{u₇\}\}

= \{X₁, X₂, X₃, X₄, X₅, X₆, X₇\}

Calculate X₁ / \{d\} = \{\{u₁\}\}, so u₁ is selected and Uₚ := \{u₁\}.

Calculate X₂ / \{d\} = \{\{u₁, u₂\}\}, so u₂ is selected and Uₚ := \{u₁, u₂\}.

Calculate X₃ / \{d\} = \{\{u₁, u₆\}\}, so u₃ is selected and Uₚ := \{u₁, u₂, u₃\}.

Calculate X₄ / \{d\} = \{\{u₅\}\}, so u₄ is selected and Uₚ := \{u₁, u₂, u₃, u₅\}.

Calculate X₅ / \{d\} = \{\{u₆\}\}, so u₅ is selected and Uₚ := \{u₁, u₂, u₃, u₄, u₅\}.

Calculate X₆ / \{d\} = \{\{u₇\}\}, so u₆ is selected and Uₚ := \{u₁, u₂, u₃, u₄, u₅, u₆\}.

Calculate X₇ / \{d\} = \{\{u₈\}\}, so u₇ is selected and Uₚ := \{u₁, u₂, u₃, u₄, u₅, u₆, u₇\}.

Calculate X₈ / \{d\} = \{\{u₉\}\}, so u₈ is selected and Uₚ := \{u₁, u₂, u₃, u₄, u₅, u₆, u₇, u₈\}.

Calculate X₉ / \{d\} = \{\{u₉\}\}, so u₉ is selected and Uₚ := \{u₁, u₂, u₃, u₄, u₅, u₆, u₇, u₈, u₉\}.

In the first set-valued decision table DS = (U, AT ∪ \{d\}) and representative set-valued decision table DSₚ = (Uₚ, AT ∪ \{d\}), we prove the following lemma:

**Lemma 2.** If uₚ ∈ U is a representative object in DS = (U, AT ∪ \{d\}) such that \(\partial_B\{uₚ\} \neq \partial_{AT}\{uₚ\}\) where \(B \subset AT\), then \(\partial_B\{uₚ\} \neq \partial_{AT}\{uₚ\}\) in DSₚ = (Uₚ, AT ∪ \{d\}) where uₚ ∈ Uₚ.

**Proof:** On DS = (U, AT ∪ \{d\}). from assumption \(\partial_B\{uₚ\} \neq \partial_{AT}\{uₚ\}\) we have \(T_B\{uₚ\} \neq T_{AT}\{uₚ\}\). Suppose that \(T_B\{uₚ\} = T_{AT}\{uₚ\}\) Y, then there exists \(y \in Y\) such that \(d\{y\} \in \partial_{AT}\{uₚ\}\), we have \(y \notin \partial_{AT}\{uₚ\}\).

1) If y is a representative object, it means that \(y = y_p\), then on DSₚ = (Uₚ, AT ∪ \{d\}), \(d\{y\} \notin \partial_{AT}\{uₚ\}\) and \(d\{y\} \in \partial_B\{uₚ\}\), so we can conclude that \(\partial_B\{uₚ\} \neq \partial_{AT}\{uₚ\}\).

2) If y is not a representative object, suppose that \(y_p\) is a representative object of an equivalence class which contains y, based on the method of building representative object set, we have \(d\{y_p\} = d\{y\}\), from \(d\{y\} \notin \partial_{AT}\{uₚ\}\), on DSₚ = (Uₚ, AT ∪ \{d\}), we have \(d\{y\} \notin \partial_{AT}\{uₚ\}\).
In addition, on $DS=(U,AT\cup\{d\})$ we have $d(y_p)=d(y)$, from $d(y)\in \partial_B(u_p)$ we have $d(y_p)\in \partial_B(u_p)$.

Thus, on $DS_p=(U_p,AT\cup\{d\})$ we have $d(y_p)\in \partial_B(u_p)$. (ii).

From (i) and (ii) we conclude that $\partial_B(u_p)\neq \partial_AT(u_p)$.

Thus, both two situation 1) and 2) we have $\partial_B(u_p)\neq \partial_AT(u_p)$ on $DS_p=(U_p,AT\cup\{d\})$.

Next, we prove that the reduct of initial set-valued decision system and the reduct of representative set-valued decision system are the same.

Assumption $R \subseteq AT$ is a reduct of the initial set-valued decision system $DS=(U,AT\cup\{d\})$, then $\partial_B(u)\subseteq \partial_AT(u)$ for any $u \in U$ such that $\partial_B(u)\neq \partial_AT(u)$.

1) From $\partial_B(u)\subseteq \partial_AT(u)$ for any $u \in U$ on $DS=(U,AT\cup\{d\})$ we have $\partial_B(u_p)\subseteq \partial_AT(u_p)$ for any $u_p \in U_p$ on $DS_p=(U_p,AT\cup\{d\})$.

2) Without generality, assume that $B \subseteq R$ and exist $u \in U$ on $DS=(U,AT\cup\{d\})$ such that $\partial_B(u)\neq \partial_AT(u)$.

If $u$ is a representative object then $u=u_p$ and $\partial_B(u_p)\neq \partial_AT(u_p)$ on $DS=(U,AT\cup\{d\})$, according to lemma 2 we have $\partial_B(u_p)\neq \partial_AT(u_p)$ on $DS_p=(U_p,AT\cup\{d\})$ (i).

If $u$ is not a representative object then on $DS=(U,AT\cup\{d\})$, assume that $u_p$ is a representative object of equivalence class $[u_p]_{AT}$ which contains $u$ and $u_p$, then $[u_p]_{AT}=[u]_{AT}$. Because of $B \subseteq R \subseteq AT$, so from $[u_p]_{AT}=[u]_{AT}$ we have $[u_p]_B=[u]_B$. From $[u_p]_{AT}=[u]_{AT}$ we have $T_AT(u_p)=T_AT(u)$, thus $\partial_AT(u_p)=\partial_AT(u)$. From $[u_p]_B=[u]_B$, by similar method we have $\partial_B(u_p)=\partial_B(u)$. According to the assumption, $\partial_B(u)\neq \partial_AT(u)$ so we have $\partial_B(u_p)\neq \partial_AT(u_p)$ on $DS=(U,AT\cup\{d\})$, according to lemma 2, we have $\partial_B(u_p)\neq \partial_AT(u_p)$ on $DS_p=(U_p,AT\cup\{d\})$. (ii)

Thus, both two situations (i) and (ii) we have $\partial_B(u_p)\neq \partial_AT(u_p)$ on $DS_p=(U_p,AT\cup\{d\})$, so we can conclude that there exists $B \subseteq R$ such that $\partial_B(u_p)\neq \partial_AT(u_p)$.

From 1) and 2) according to the definition, we have $R \subseteq AT$ is a reduct of representative set-valued decision system $DS_p=(U_p,AT\cup\{d\})$.

V. Conclusion

In this paper, we presented a method of selecting of representative object set for attribute reduction in set-valued decision system. The authors evaluated and researched relation between reduct’ definitions in set-valued decision system. The authors relised that reduct is similar in term of consistent decision table. For inconsistent decision table, the authors separated reducts into three group and presented the relation of reduct between groups. In order to create new and efficiency methods, a meaning result help us deeply understand characteristic of reduct’s definition in set-valued decision system and is a theory foundation to evaluate attribute reduction methods. Based on the results, further research is researching an alternative value of evaluating an decision reduct set’s efficiency in set-valued decision system in order to complete problem.

References

[8] Shifei D., Hao D., Research and Development of Attribute Reduction Algorithm Based on Rough Set.
Relation between definition of Reduct in Set-valued Decision System


Relations in Disjunctive Set-Valued Ordered
Information Systems, International Journal of
Information Technology & Decision Making, Vol. 9,
No. 1, 2010, pp. 9–33.

[12] Yuhua Qian, Chuangyin Dang, Jiye Liang, and Dawei
Inf. Sci. 179, 16 (July 2009), 2809–2832.

compression with homomorphism in covering
information systems, International Journal of

[14] 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,
2012, pp. 620–635.

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.

Thanh Dai Tran was born in Viet Nam on June 18,
1983. He is a lecturer in the Department of Information
Technology at the University of Economic and
Technical Industries, VietNam.