Certain, Generalized Decision, and Membership Distribution Reducts Versus Functional Dependencies in Incomplete Systems*

Marzena Kryszkiewicz

Institute of Computer Science, Warsaw University of Technology
Nowowiejska 15/19, 00-665 Warsaw, Poland
mkr@ii.pw.edu.pl

Abstract. An essential notion in the theory of Rough Sets is a reduct, which is a minimal set of conditional attributes that preserves a required classification feature, e.g. respective values of an original or modified decision attribute. Certain decision reducts, generalized decision reducts, and membership distribution reducts belong to basic types of Rough Sets reducts. In our paper, we prove that reducts of these types are sets of conditional attributes functionally determining respective modifications of a decision attribute both in complete and incomplete information systems. However, we also prove that, unlike in the case of complete systems, the reducts in incomplete systems are not guaranteed to be minimal sets of conditional attributes that functionally determine respective modifications of the decision attribute.

1 Introduction

Rough Sets theory defines reducts in a decision table as minimal sets of conditional attributes preserving the required classification feature [10]. The research devoted to reducts referred mostly to complete systems in which all attribute values were known. In this paper, we first revisit the results for certain decision, generalized decision, and membership distribution reducts, which belong to basic types of Rough Set reducts. Next, we examine properties of reducts of these types in incomplete systems in which values of attributes may be missing. As a result, we prove that reducts of these types are sets of conditional attributes functionally determining respective modifications of a decision attribute both in complete and incomplete information systems. However, we also prove that, unlike in the case of complete systems, the reducts in incomplete systems are not guaranteed to be minimal sets of conditional attributes that functionally determine respective modifications of the decision attribute.

The layout of the paper is as follows: In Section 2, we recall basic Rough Set notions and provide their properties. A notion of a functional dependency is recalled in Section 3. In Section 4, we systematically revisit the relationship

* Research has been supported by grant No 3 T11C 002 29 received from Polish Ministry of Education and Science.

M. Kryszkiewicz et al. (Eds.): RSEISP 2007, LNAI 4585, pp. 162–174, 2007.

between functional dependencies and generalized decision reducts, membership distribution reducts, and certain decision reducts in complete decision tables. The main part of our contribution is presented in Section 5, where we examine this relationship in incomplete decision tables. In Section 6, we conclude our results.

2 Basic Notions and Properties of Rough Sets

2.1 Information Systems

An *information system* (*IS*) is a pair $S = (O, AT)$, where O is a non-empty finite set of *objects* and AT is a non-empty finite set of *attributes*, such that $a : O \rightarrow V_a$ for any $a \in AT$, where V_a is called *domain* of the attribute a. Each subset of attributes $A \subseteq AT$ determines a binary A-*indiscernibility relation* $IND(A)$,

$$IND(A) = \{(x, y) \in O \times O \mid \forall_{a \in A}\ a(x) = a(y)\}.$$

The relation $IND(A), A \subseteq AT$, is an equivalence relation and determines a partition of O, which will be denoted by π_A. Objects indiscernible with object x with regard to attribute set A in the system will be denoted by $I_A(x)$ and called A-*indiscernibility class*; that is, $I_A(x) = \{y \in O \mid (x, y) \in IND(A)\}$. Clearly, partition $\pi_A = \{I_A(x) \mid x \in O\}$.

Property 2.1.1 [10]. Let $A, B \subseteq AT$ and $x \in O$.
a) $A \subseteq B \Rightarrow I_B(x) \subseteq I_A(x)$
b) $I_{A \cup B}(x) = I_A(x) \cap I_B(x)$
c) $I_A(x) = \bigcap_{a \in A} I_a(x)$

Proposition 2.1.1. Let $A \subseteq B \subseteq AT$ and $x \in O$. $I_A(x) = \bigcup_{y \in I_A(x)} I_B(y)$.

Example 2.1.1. Table 1 describes a sample information system consisting of 10 objects and described by attributes $\{a, b, c, e, f, d\}$. Let $A = \{a, b\}$ and $B =$

Table 1. Sample DT

$x \in O$	a	b	c	e	f	d
1	1	0	0	1	1	1
2	1	1	1	1	2	1
3	0	1	1	0	3	1
4	0	1	1	0	3	2
5	0	1	1	2	2	2
6	1	1	0	2	2	2
7	1	1	0	2	2	3
8	1	1	0	2	2	3
9	1	1	0	3	2	3
10	1	0	0	3	2	3

Table 2. DT extended with d^{N}_{AT}, ∂_{AT}, μ^{AT}_d

$x \in O$	a	b	c	e	f	d	d^{N}_{AT}	∂_{AT}	$\mu^{AT}_d : < \mu^{AT}_1, \mu^{AT}_2, \mu^{AT}_3 >$
1	1	0	0	1	1	1	1	$\{1\}$	$< 1, 0, 0 >$
2	1	1	1	1	2	1	1	$\{1\}$	$< 1, 0, 0 >$
3	0	1	1	0	3	1	N	$\{1, 2\}$	$< 1/2, 1/2, 0 >$
4	0	1	1	0	3	2	N	$\{1, 2\}$	$< 1/2, 1/2, 0 >$
5	0	1	1	2	2	2	2	$\{2\}$	$< 0, 1, 0 >$
6	1	1	0	2	2	2	N	$\{2, 3\}$	$< 0, 1/3, 2/3 >$
7	1	1	0	2	2	3	N	$\{2, 3\}$	$< 0, 1/3, 2/3 >$
8	1	1	0	2	2	3	N	$\{2, 3\}$	$< 0, 1/3, 2/3 >$
9	1	1	0	3	2	3	3	$\{3\}$	$< 0, 0, 1 >$
10	1	0	0	3	2	3	3	$\{3\}$	$< 0, 0, 1 >$

$\{a, b, c, e, f\}$. $I_A(3) = \{3, 4, 5\}$, $I_B(3) = I_B(4) = \{3, 4\}$, $I_B(5) = \{5\}$. Hence, $I_A(3) = \{3, 4, 5\} = I_B(3) \cup I_B(4) \cup I_B(5)$ (see Proposition 2.1.1). $\square$

Let $X \subseteq O$ and $A \subseteq AT$. AX is defined as an *A-lower approximation* of object set X, iff $\underline{A}X = \bigcup\{Y \in \pi_A \mid Y \subseteq X\}$ (or $\underline{A}X = \{x \in O \mid I_A(x) \subseteq X\}$). $\overline{A}X$ is defined as an *A-upper approximation* of X, iff $\underline{A}X = \bigcup\{Y \in \pi_A \mid Y \cap X \neq \emptyset\}$ (or $\overline{A}X = \{x \in O \mid I_A(x) \cap X \neq \emptyset\}$). $\underline{A}X$ is the set of objects that belong to X with certainty, while $\overline{A}X$ is the set of objects that possibly belong to X.

2.2 Decision Tables

A *decision table* is an information system $DT = (O, AT \cup \{d\})$, where $d \notin AT$ is a distinguished attribute called the *decision*, and the elements of AT are called *conditions*. A *decision class* is defined as the set of all objects with the same decision value. By X_{d_i} we will denote the decision class consisting of objects the decision value of which equals d_i, where $d_i \in V_d$. Clearly, for any object x in O, $I_d(x)$ is a decision class. DT is called *consistent* if for each $I_{AT}(x) \in \pi_{AT}$ there is $I_d(x) \in \pi_d$ such that $I_{AT}(x) \subseteq I_d(x)$. Otherwise, DT is called *inconsistent*.

Proposition 2.2.1. Let $A \subseteq AT$ and $x \in X \subseteq O$. $X \subseteq I_d(x)$ iff $\exists_{y \in O}\ X \subseteq I_d(y)$.

Proof. ($\Rightarrow$) Trivial.
($\Leftarrow$) Let y be an object in O such that $X \subseteq I_d(y)$ (*). Hence, $x \in X \subseteq I_d(y)$, so $x \in I_d(y)$. Thus, $d(x) = d(y)$ (**). By (*) and (**), $X \subseteq I_d(y) = I_d(x)$. $\square$

An *A-positive region* (denoted by POS_A) in DT is defined as the union of the A-lower approximations of all decision classes, that is:

$$POS_A = \bigcup_{d_i \in V_d} \underline{A}X_{d_i}.$$

For $A = AT$, A-positive region is denoted briefly by POS.

Proposition 2.2.2. $POS_A = \{x \in O \mid I_A(x) \subseteq I_d(x)\}$.

Proof. $POS_A = \bigcup_{d_i \in V_d} \underline{A}X_{d_i} = \bigcup_{y \in O} \underline{A}I_d(y) = \bigcup_{y \in O} \{x \in O \mid I_A(x) \subseteq I_d(y)\} = /^*$ by Proposition 2.2.1 $^*/ = \bigcup_{y \in O} \{x \in O \mid I_A(x) \subseteq I_d(x)\} = \{x \in O \mid I_A(x) \subseteq I_d(x)\}$. $\square$

One can note that the positive region contains all objects in O about which we are certain that they belong to the decision classes determined by their decision values. An *A-negative region* (NEG_A) is defined as the set of all objects in O that do not belong to POS_A. In the sequel, NEG_{AT} will be denoted briefly by NEG. Clearly, DT is consistent iff $NEG = \emptyset$ (or $POS = O$).

For the sake of later use, we introduce a notion of an *A-derivable decision attribute* for an object $x \in O$, which we denote by $d_A^{\mathrm{N}}(x)$ and define as follows: $d_A^{\mathrm{N}}(x) = d(x)$ if $x \in POS_A$, and $d_A^{\mathrm{N}}(x) = \mathrm{N}$ otherwise. Clearly, all objects with value N of d_{AT}^{N} belong to NEG; all other objects belong to POS.

The notion of the negative region may be too vague in some applications. Looking at Table 1, one may note that objects 3 and 4, which are indiscernible with respect to $AT = \{a, b, c, e, f\}$, may belong to the decision classes X_{d_1} or X_{d_2}, but certainly do not belong the decision class X_{d_3}.

A notion of a generalized decision allows us to specify this knowledge. An A *generalized decision* for object x in DT (denoted by $\partial_A(x)$), $A \subseteq AT$, is defined as the set of all decision values of all objects indiscernible with x on A; i.e. [13]:

$$\partial_A(x) = \{d(y) \mid y \in I_A(x)\}.$$

Property 2.2.1. Let $x \in O$ and $A, B \subseteq AT$. If $A \subseteq B$, then $\partial_B(x) \subseteq \partial_A(x)$.

For $A = AT$, an *A-generalized decision* will be also called briefly a *generalized decision*. The generalized decision informs on decision classes to which an object may belong. One may additionally be interested in the degree in which the objects may belong to these classes. An *A-membership function*: $\mu^A_{d_i} : O \rightarrow [0, 1]$, $A \subseteq AT$, is defined as follows [15]:

$$\mu^A_{d_i}(x) = \frac{\mid I_A(x) \cap X_{d_i} \mid}{\mid I_A(x) \mid}.$$

An *A-membership distribution function*: $\mu^A_d : O \rightarrow [0, 1]^n$, $A \subseteq AT, n = \mid V_d \mid$, is defined as follows [15]:

$$\mu^A_d(x) = (\mu^A_{d_1}(x), \ldots, \mu^A_{d_n}(x)), \text{where } \{d_1, \ldots, d_n\} = V_d.$$

The values of the derivable decision attribute, generalized decision and membership distribution function for objects in DT from Table 1 are shown in Table 2.

2.3 Certain Decision, Generalized Decision, and Membership Distribution Reducts

A *reduct* is an essential notion in the Rough Set theory. In this paper, we will focus on three types of reducts, namely, on certain decision, generalized decision, and membership distribution reducts. Below, we recall their definitions:

A set of attributes $A \subseteq AT$ is a *certain decision reduct* of DT iff A is a minimal set such that

$$\forall_{x \in POS}\ I_A(x) \subseteq I_d(x).$$

$A \subseteq AT$ is a *generalized decision reduct* of DT iff A is a minimal set such that

$$\forall_{x \in O}\ \partial_A(x) = \partial_{AT}(x).$$

$A \subseteq AT$ is a *μ-decision reduct* (or *membership distribution reduct*) of DT iff A is a minimal set such that

$$\forall_{x \in O}\ \mu^A_d(x) = \mu^{AT}_d(x).$$

In general, for each certain decision reduct A, there is a superset of A which is a generalized decision reduct, and for each generalized decision reduct B, there is a superset of B which is a μ-decision reduct [6],[7]. In the Rough Set literature, one can also find definitions of other types of reducts. To the most important ones, we did not introduce, belong possible, approximate and μ-reducts. It has been proved in [6],[7] that the set of possible reducts as well as the set of approximate reducts equals the set of generalized decision reducts, and the set of μ-reducts of DT equals the set of μ-decision reducts. These and other types of reducts were also discussed e.g. in [1],[8-19].

3 Functional Dependencies

Let A and B be sets of attributes in an information system. $A \to B$ is defined a *functional dependency* (or A is defined to *determine B functionally*) if $\forall_{x \in O}\ I_A(x) \subseteq I_B(x)$. $A \to B$ is defined a *minimal functional dependency* if it is a functional dependency and $\forall_{C \subset A}\ C \to B$ is not a functional dependency.

Example 3.1. Let us consider the information system in Table 1. $\{ce\} \to \{a\}$ is a functional dependency, nevertheless, $\emptyset \to \{a\}$, $\{c\} \to \{a\}$ and $\{e\} \to \{a\}$ are not. Hence, $\{ce\} \to \{a\}$ is a minimal functional dependency. $\square$

4 Reducts and Minimal Functional Dependencies

In this section, we prove that generalized decision, membership distribution, and certain decision reducts are minimal sets of conditional attributes in decision table DT which functionally determine the generalized decision ∂_{AT}, membership distribution μ_d^{AT}, and derivable decision attribute $d^{\mathrm{N}}_{AT}(x)$, respectively.

4.1 Generalized Decision Reducts and Minimal Functional Dependencies

Since generalized decision reducts are based on the notion of a generalized decision, we first examine the relationship between this notion and a functional dependency.

Lemma 4.1.1. Let $A \subseteq AT$. The following statements are equivalent:
a) $\forall_{x \in O}\ \partial_A(x) = \partial_{AT}(x)$
b) $\forall_{x \in O}\ \forall_{y \in I_A(x)} \partial_{AT}(y) = \partial_{AT}(x)$
c) $\forall_{x \in O}\ I_A(x) \subseteq I_{\partial_{AT}}(x)$
d) $A \to \{\partial_{AT}\}$ is a functional dependency

Proof. Ad a $\Rightarrow$ b) (by contradiction). Let $\forall_{z \in O}\ \partial_A(z) = \partial_{AT}(z)$ $(^*)$, $x \in O$, $y \in I_A(x)$ $(^{**})$ and $\partial_{AT}(y) \neq \partial_{AT}(x)$. By $(^*)$, $\partial_A(x) = \partial_{AT}(x)$, $\partial_A(y) = \partial_{AT}(y)$, and by $(^{**})$, $\partial_A(x) = \partial_A(y)$. Hence, $\partial_{AT}(x) = \partial_A(x) = \partial_A(y) = \partial_{AT}(y)$. Thus, we conclude, $\partial_{AT}(x) = \partial_{AT}(y)$, which contradicts the assumption. Ad a $\Leftarrow$ b) Let $x \in O$ and $\forall_{y \in I_A(x)}\ \partial_{AT}(y) = \partial_{AT}(x)$ $(^*)$. $\partial_A(x) = \bigcup_{y \in I_A(x)} \{d(y)\}$ $\subseteq$ /* $d(y) \in \partial_{AT}(y)$ for any object y */ $\bigcup_{y \in I_A(x)} \partial_{AT}(y)$ = /* by $(^*)$ */ = $\bigcup_{y \in I_A(x)}\ \partial_{AT}(x) = \partial_{AT}(x)$. Hence, $\partial_A(x) \subseteq \partial_{AT}(x)$ $(^{**})$. On the other hand,

by Property 2.1.1, $\partial_{AT}(x) \subseteq \partial_A(x)$ (***). By (**) and (***), we conclude, $\partial_A(x) = \partial_{AT}(x)$.
Ad b $\Leftrightarrow$ c $\Leftrightarrow$ d) Trivial. □

Proposition 4.1.1. $AT \rightarrow \{\partial_{AT}\}$ is a functional dependency.

Proof. The formula $\forall_{x \in O}\ \partial_{AT}(x) = \partial_{AT}(x)$ is trivially true. Hence, and by Lemma 4.1.1a,d, $AT \rightarrow \{\partial_{AT}\}$ is a functional dependency. □

Theorem 4.1.1. Let $A \subseteq AT$. A is a generalized decision reduct of DT iff $A \rightarrow \{\partial_{AT}\}$ is a minimal functional dependency.

Proof. A is a generalized decision reduct of DT iff /* by definition of a generalized decision reduct /* $\forall_{x \in O}\ \partial_A(x) = \partial_{AT}(x)$ and there is no proper subset $C \subset A$ such that $\forall_{x \in O}\ \partial_C(x) = \partial_{AT}(x)$ iff /* by Lemma 4.1.1a,d /* $A \rightarrow \{\partial_{AT}\}$ is functional and there is no proper subset $C \subset A$ such that $C \rightarrow \{\partial_{AT}\}$ is functional iff $A \rightarrow \{\partial_{AT}\}$ is a minimal functional dependency. □

Theorem 4.1.1 corresponds to the result obtained in [13].

4.2 μ-Decision Reducts and Minimal Functional Dependencies

As μ-decision reducts are based on the notion of a membership distribution function, we first examine the relationship between this notion and a functional dependency.

Lemma 4.2.1. Let $A \subseteq AT$. The following statements are equivalent:
a) $\forall_{x \in O}\ \mu_d^A(x) = \mu_d^{AT}(x)$
b) $\forall_{x \in O}\ \forall_{y \in I_A(x)}\ \mu_d^{AT}(y) = \mu_d^{AT}(x)$
c) $\forall_{x \in O}\ I_A(x) \subseteq I_{\mu_d^{AT}}(x)$
d) $A \rightarrow \{\mu_d^{AT}\}$ is a functional dependency

Proof. Ad a $\Rightarrow$ b) (by contradiction). Let $\forall_{z \in O}\ \mu_d^A(z) = \mu_d^{AT}(x)$ (*), $x \in O$, $y \in I_A(x)$ (**) and $\mu_d^{AT}(y) \neq \mu_d^{AT}(x)$. By (*), $\mu_d^A(x) = \mu_d^{AT}(x)$, $\mu_d^A(y) = \mu_d^{AT}(y)$, and by (**), $\mu_d^A(x) = \mu_d^A(y)$. Hence, $\mu_d^{AT}(x) = \mu_d^A(x) = \mu_d^A(y) = \mu_d^{AT}(y)$. Thus, we conclude, $\mu_d^{AT}(x) = \mu_d^{AT}(y)$, which contradicts the assumption. Ad a $\Leftarrow$ b) Let $x \in O$ and $\forall_{y \in I_A(x)}\ \mu_d^{AT}(y) = \mu_d^{AT}(x)$ (or equivalently, $\mu_{d_i}^{AT}(y) = \mu_{d_i}^{AT}(x)$ for all $d_i \in V_d$) (*). Let d_i be an arbitrary decision value in V_d, $\mu_{d_i}^{AT}(x) = \varepsilon$, and $I_A(x) = I_1 \cup \ldots I_l$, where $I_1, \ldots, I_l$ are distinct (mutually exclusive) classes in π_{AT}. Clearly, for each class $I_j, j = 1..l$, there is an object $y \in I_A(x)$ such that $I_j = I_{AT}(y)$ and $\mid I_j \cap X_{d_i} \mid / \mid I_j \mid = \mid I_{AT}(y) \cap X_{d_i} \mid / \mid I_{AT}(y) \mid = \mu_{d_i}^{AT}(y) =$ /* by (*) */ $= \mu_{d_i}^{AT}(x)$. Hence, $\forall_{j=1..l} \mid I_j \cap X_{d_i} \mid / \mid I_j \mid = \mu_{d_i}^{AT}(x) = \varepsilon$, so $\forall_{j=1..l} \mid I_j \cap X_{d_i} \mid = \varepsilon \times \mid I_j \mid$ (**). Now, $\mu_{d_i}^A(x) = \mid I_A(x) \cap X_{d_i} \mid / \mid I_A(x) \mid = \mid (\bigcup_{j=1..l} I_j) \bigcap X_{d_i} \mid / \mid \bigcup_{j=1..l} I_j \mid = (\sum_{j=1..l} \mid I_j \cap X_{d_i} \mid) / (\sum_{j=1..l} \mid I_j \mid) = (\sum_{j=1..l} \varepsilon \times \mid I_j \mid) / (\sum_{j=1..l} \mid I_j \mid) = \varepsilon = \mu_{d_i}^{AT}(x)$. Hence, $\mu_{d_i}^A(x) = \mu_{d_i}^{AT}(x)$ (**).

As d_i was chosen arbitrarily, we may generalize (**) for all values d_i in V_d. In consequence, we conclude, $\mu_d^A(x) = \mu_d^{AT}(x)$.
Ad b $\Leftrightarrow$ c $\Leftrightarrow$ d) Trivial. □

Proposition 4.2.1. $AT \rightarrow \{\mu_d^{AT}\}$ is a functional dependency.

Proof. Analogical to the proof of Proposition 4.1.1; by Lemma 4.2.1a,d. □

Theorem 4.2.1. Let $A \subseteq AT$. A is a μ-decision reduct of DT iff $A \rightarrow \{\mu_d^{AT}\}$ is a minimal functional dependency.

Proof. Analogical to the proof of Theorem 4.1.1; follows from the definitions of a μ-decision reduct and minimal functional dependency, and Lemma 4.2.1a,d. □

Theorem 4.2.1 corresponds to the result reported in [16].

4.3 Certain Decision Reducts and Minimal Functional Dependencies

Certain decision reducts preserve the positive region. Let us thus start with investigating the consequences of (non-) belonging to POS.

Property 4.3.1. Let $x \in O$. The following statements are equivalent:
a) $x \in POS$
b) $I_{AT}(x) \subseteq I_d(x)$
c) $I_{AT}(x) \subseteq POS$

Proof. Ad (a $\Leftrightarrow$ b) By Proposition 2.2.2.
Ad (a $\Rightarrow$ c) Let $x \in POS$. Then by Proposition 2.2.2, $I_{AT}(x) \subseteq_d (x)$ (*). Since $\forall_{y \in I_{AT}(x)}\ I_{AT}(y) = I_{AT}(x)$, then (*) can be rewritten as $\forall_{y \in I_{AT}(x)}\ I_{AT}(y) \subseteq I_d(x)$. Hence, by Proposition 2.2.1, $\forall_{y \in I_{AT}(x)}\ I_{AT}(y) \subseteq I_d(y)$. Thus, by Proposition 2.2.2, $\forall_{y \in I_{AT}(x)}\ y \in POS$, so $I_{AT}(x) \subseteq POS$.
Ad (a $\Leftarrow$ c) Trivial. □

Property 4.3.2. Let $x \in O$. The following statements are equivalent:
a) $x \notin POS$
b) $I_{AT}(x) \not\subseteq I_d(x)$
c) $I_{AT}(x) \subseteq O \backslash POS$

Proof. Ad (a $\Leftrightarrow$ b) Follows from Property 4.3.1.
Ad (b $\Rightarrow$ c) Let $I_{AT}(x) \not\subseteq I_d(x)$. Then, by Proposition 2.2.1, $\neg\exists_{y \in O}\ I_{AT}(x) \subseteq I_d(y)$. Hence, $\forall_{y \in I_{AT}(x)}\ I_{AT}(x) \not\subseteq I_d(y)$. Since $\forall_{y \in I_{AT}(x)}\ I_{AT}(y) = I_{AT}(x)$, then $\forall_{y \in I_{AT}(x)}\ I_{AT}(y) \not\subseteq I_d(y)$. Thus by Property 4.3.1, $\forall_{y \in I_{AT}(x)}\ y \in O \backslash POS$. Therefore, $I_{AT}(x) \subseteq O \backslash POS$.
Ad (b $\Leftarrow$ c) Let $I_{AT}(x) \subseteq O \backslash POS$. Hence, $I_{AT}(x) \not\subseteq POS$. Then, by Property 4.3.1, $I_{AT}(x) \not\subseteq I_d(x)$. □

By Property 4.3.1, if object x belongs to POS, then AT-indiscernibility class of this object is contained in POS, and all objects in this class have the same decision value as x does. By Property 4.3.2, if x does not belong to POS, then AT-indiscernibility class of this object is contained in the negative region.

Lemma 4.3.1. Let $A \subseteq AT$ and $\forall_{y \in O}\ I_{AT}(y) \subseteq I_d(y) \Rightarrow I_A(y) \subseteq I_d(y)$. Then:
a) $\forall_{x \in O}\ I_{AT}(x) \subseteq I_d(x) \Rightarrow I_A(x) \subseteq POS$
b) $\forall_{x \in POS}\ I_A(x) \subseteq POS$
c) $\forall_{x \in O}\ I_{AT}(x) \not\subseteq I_d(x) \Rightarrow I_A(x) \subseteq O \backslash POS$
d) $\forall_{x \in O \backslash POS}\ I_A(x) \subseteq O \backslash POS$

Proof. Let $A \subseteq AT$ and $\forall_{y \in O}\ I_{AT}(y) \subseteq I_d(y) \Rightarrow I_A(y) \subseteq I_d(y)$ (*).
Ad a) Let x be an object such that $I_{AT}(x) \subseteq I_d(x)$. By Proposition 2.1.1 and (*) we conclude, $\bigcup_{y \in I_A(x)}\ I_{AT}(y) = I_A(x) \subseteq I_d(x)$. Hence and by Proposition 2.2.1, $\forall_{y \in I_A(x)}\ I_{AT}(y) \subseteq I_d(y)$. Thus, by Property 4.3.1, $\forall_{y \in I_A(x)}\ I_{AT}(y) \subseteq POS$. Having this result in mind and taking into account Proposition 2.1.1, we conclude $I_A(x) = \bigcup_{y \in I_A(x)}\ I_{AT}(y) \subseteq POS$.

Ad b) Follows from Lemma 4.3.1a and Property 4.3.1.
Ad c) (by contradiction). Let x be an object such that $I_{AT}(x) \not\subseteq I_d(x)$ (**) and $I_A(x) \not\subseteq O\backslash POS$. By Proposition 2.1.1, we conclude: $\bigcup_{y \in I_A(x)}\ I_{AT}(y) \not\subseteq O\backslash POS$. Hence, $\exists_{y \in I_A(x)}\ y \in POS$. Thus, by Property 4.3.1, $\exists_{y \in I_A(x)}\ I_{AT}(y) \subseteq I_d(y)$. By (*) we conclude: $\exists_{y \in I_A(x)}\ I_A(y) \subseteq I_d(y)$. Since, $I_A(y) = I_A(x)$ for any $y \in I_A(x)$, then we may infer $\exists_{y \in I_A(x)}\ I_A(x) \subseteq I_d(y)$. Now, by Proposition 2.2.1, we may derive, $I_A(x) \subseteq I_d(x)$. Since $I_{AT}(x) \subseteq I_A(x)$ (by Property 2.1.1a), we conclude, $I_{AT}(x) \subseteq I_d(x)$. This contradicts the assumption (**).
Ad d) Follows from Lemma 4.3.1c and Property 4.3.2. □

Lemma 4.3.2. Let $A \subseteq AT$. The following statements are equivalent:
a) $\forall_{x \in POS}\ I_A(x) \subseteq I_d(x)$
b) $\forall_{x \in O}\ I_A(x) \subseteq I_{d^{\mathrm{N}}_{AT}}(x)$
c) $A \rightarrow \{d^{\mathrm{N}}_{AT}\}$ is a functional dependency

Proof. Ad a $\Rightarrow$ b) Let $\forall_{x \in POS}\ I_A(x) \subseteq I_d(x)$(*). Hence, by Property 4.3.1, $\forall_{x \in O}\ I_{AT}(x) \subseteq I_d(x) \Rightarrow I_A(x) \subseteq I_d(x)$. Thus, by Lemma 4.3.1d, $\forall_{x \in O\backslash POS}\ I_A(x) \subseteq O\backslash POS$ (**). Since $d^{\mathrm{N}}_{AT}(x) = \mathrm{N}$ for all and only $x \in O\backslash POS$, then $\forall_{x \in O\backslash POS}\ I_{d^{\mathrm{N}}_{AT}}(x) = O\backslash POS$. Hence, (**) can be rewritten as $\forall_{x \in O\backslash POS}\ I_A(x) \subseteq I_{d^{\mathrm{N}}_{AT}}(x)$ (***). In addition, since $d^{\mathrm{N}}_{AT}(x) = d(x)$ for $x \in POS$, then (*) can be rewritten as $\forall_{x \in POS}\ I_A(x) \subseteq I_{d^{\mathrm{N}}_{AT}}(x)$(****). Thus, by (***) and (****), $\forall_{x \in O}\ I_A(x) \subseteq I_{d^{\mathrm{N}}_{AT}}(x)$.
Ad a $\Leftarrow$ b) Let $\forall_{x \in O}\ I_A(x) \subseteq I_{d^{\mathrm{N}}_{AT}}(x)$. Then, by definition of d^{N}_{AT}, $\forall_{x \in POS}\ I_A(x) \subseteq I_{d^{\mathrm{N}}_{AT}}(x) = I_{d_{AT}}(x)$.
Ad b $\Leftrightarrow$ c) Trivial. □

Having in mind properties of the positive region (Proposition 2.2.2), definition of a certain decision reduct and Lemma 4.3.2, we offer Proposition 4.3.1 and Theorem 4.3.1, in which we express the relationship between certain decision reducts and functional dependencies.

Proposition 4.3.1. $AT \rightarrow \{d^{\mathrm{N}}_{AT}\}$ is a functional dependency.
Proof. By Proposition 2.2.2, $\forall_{x \in POS}\ I_{AT}(x) \subseteq I_d(x)$. Hence and by Lemma 4.3.2a,c, $AT \rightarrow \{d^{\mathrm{N}}_{AT}\}$ is a functional dependency. □

Theorem 4.3.1. Let $A \subseteq AT$ A is a certain decision reduct of DT iff $A \rightarrow \{d^{\mathrm{N}}_{AT}\}$ is a minimal functional dependency.

Proof. Analogical to the proof of Theorem 4.1.1; follows from the definition of a certain decision reduct, definition of a minimal functional dependency and Lemma 4.3.2a,c. □

Theorem 4.3.1 corresponds to the result presented in [14].

5 Reducts and Functional Dependencies Under Incompleteness

It may happen that some of attribute values for an object are missing in an information system. The system in which values of all attributes for all objects from O are known is called *complete*, otherwise it is called *incomplete*. Further on, we will denote missing value by *. We will also assume that an object $x \in O$ possesses exactly one value for each attribute in AT, in reality. Thus, if the value of an attribute a is missing, then we conclude that the real value is one from the set $V_a \backslash \{^*\}$. Hence, an object with $a(x) = {}^*$ is likely to be $\{a\}$-indiscernible in reality with all other objects in O. The indiscernibility relation, nevertheless, would treat this object as indiscernible only with objects for which the value of attribute a is unknown, which seems incorrect. In [2-5], we have introduced and discussed a notion of a similarity relation in order to deal with the incompleteness. In this section, we examine the dependency between similarity-based certain decision, generalized decision, and μ-decision reducts and respective modification of the decision attribute.

5.1 Basic Notions Under Incompleteness

In Section 5, we consider an incomplete decision table $IDT = (O, AT \cup \{d\})$ that admits unknown values only for attributes in AT. A *similarity relation* wrt. $A \subseteq AT$ is denoted by $SIM(A)$, and is defined as follows:

$$SIM(A) = \{(x, y) \in O \times O \mid \forall_{a \in A}\ a(x) = a(y) \text{ or } a(x) = {}^* \text{ or } a(y) = {}^*\}.$$

The similarity relation is reflexive and symmetric, but may not be transitive. The set of objects similar with object x wrt. attribute set A in IDT is denoted by $S_A(x)$ and called A-*similarity class*; that is, $S_A(x) = \{y \in O \mid (x, y) \in SIM(A)\}$.

Example 5.1.1. Table 3 presents a sample incomplete decision table $IDT = (O, AT \cup \{d\})$, where $AT = \{a, b\}$. The similarity classes of objects 1 and 5 wrt. AT, $\{b\}$, and $\emptyset$ are as follows: $S_{AT}(1) = \{1\}, S_{\{b\}}(1) = \{1, 5\}$, $S_\emptyset(1) = \{1, 2, 3, 4, 5, 6, 7, 8\}$, $S_{AT}(5) = \{5, 6\}$, $S_{\{b\}}(5) = S_\emptyset(5) = \{1, 2, 3, 4, 5, 6, 7, 8\}$. □

Table 3. $IDT = (O, AT \cup \{d\})$, where $AT = \{a, b\}$, extended with modified decisions

$x \in O$	a	b	d	d^N_{AT}	$d^N_{\{b\}}$	$d^N_\emptyset$	∂_{AT}	$\partial_{\{b\}}$	$\partial_\emptyset$	μ_d^{AT}	$\mu_d^{\{b\}}$	$\mu_d^\emptyset$
1	1	1	1	1	N	N	$\{1\}$	$\{1,3\}$	$\{1,2,3\}$	$<1,0,0>$	$<1/2,0,1/2>$	$<1/8,2/8,5/8>$
2	2	3	2	N	N	N	$\{2,3\}$	$\{2,3\}$	$\{1,2,3\}$	$<0,2/3,1/3>$	$<0,2/4,2/4>$	$<1/8,2/8,5/8>$
3	2	3	2	N	N	N	$\{2,3\}$	$\{2,3\}$	$\{1,2,3\}$	$<0,2/3,1/3>$	$<0,2/4,2/4>$	$<1/8,2/8,5/8>$
4	2	3	3	N	N	N	$\{2,3\}$	$\{2,3\}$	$\{1,2,3\}$	$<0,2/3,1/3>$	$<0,2/4,2/4>$	$<1/8,2/8,5/8>$
5	3	*	3	3	N	N	$\{3\}$	$\{1,2,3\}$	$\{1,2,3\}$	$<0,0,1>$	$<1/8,2/8,5/8>$	$<1/8,2/8,5/8>$
6	3	4	3	3	3	N	$\{3\}$	$\{3\}$	$\{1,2,3\}$	$<0,0,1>$	$<0,0,1>$	$<1/8,2/8,5/8>$
7	4	5	3	3	3	N	$\{3\}$	$\{3\}$	$\{1,2,3\}$	$<0,0,1>$	$<0,0,1>$	$<1/8,2/8,5/8>$
8	5	6	3	3	3	N	$\{3\}$	$\{3\}$	$\{1,2,3\}$	$<0,0,1>$	$<0,0,1>$	$<1/8,2/8,5/8>$

Property 5.1.1. Let $A, B \subseteq AT$ and $x \in O$.
a) $I_A(x) \subseteq S_A(x)$
b) $\forall_{y \in I_A(x)}\ S_A(y) = S_A(x)$
c) $A \subseteq B \Rightarrow S_B(x) \subseteq S_A(x)$

In Table 4, we provide definitions of similarity-based Rough Sets notions, we use throughout this section. Table 3 illustrates d_A^{N}, ∂_A, and μ_d^A, where $A \subseteq AT$.

Let $A \subseteq AT$. A is a *certain decision reduct* of IDT iff A is a minimal set such that $\forall_{x \in POS}\ S_A(x) \subseteq I_d(x)$. A is a *generalized decision reduct* of IDT iff A is a minimal set such that $\forall_{x \in O}\ \partial_A(x) = \partial_{AT}(x)$. A is a *μ-decision reduct* of IDT iff A is a minimal set such that $\forall_{x \in O}\ \mu_d^A(x) = \mu_d^{AT}(x)$.

The definition of a (minimal) functional dependency in an incomplete system remains the same as in the case of a complete system (see Section 3).

Table 4. Similarity based Rough Sets notions

notion	definition	notion	definition
$\underline{A}X$	$\{x \in O \mid S_A(x) \subseteq X\}$;	$d_A^{\mathrm{N}}(x)$	$d(x)$ if $x \in POS_A$, and N otherwise;
$\overline{A}X$	$\{x \in O \mid S_A(x) \cap X \neq \emptyset\}$;	$\partial_A(x)$	$\{d(y) \mid y \in S_A(x)\}$;
POS_A	$\underline{A}X_{d_1} \cup \ldots \cup \underline{A}X_{d_n}$;	$\mu {d_i}^A(x)$	$\mid S_A(x) \cap X_{d_i} \mid / \mid S_A(x) \mid$;
POS	POS_{AT};	$\mu_d^A(x)$	$(\mu_{d_1}^A(x), \ldots, \mu_{d_n}^A(x))$.

5.2 Generalized Decision Reducts and Functional Dependencies Under Incompleteness

Lemma 5.2.1. Let $A \subseteq AT$ and $x \in O$. $\forall_{y \in I_A(x)}\ \partial_A(y) = \partial_A(x)$.

Proof. Let $y \in I_A(x)$. By definition, $\partial_A(y) = \{d(z) \mid z \in S_A(y)\} = /^*$ by Property 5.1.1b $^*/ = \{d(z) \mid z \in S_A(x)\} = \partial_A(x)$. □

Proposition 5.2.1. Let $A \subseteq AT$. $A \rightarrow \{\partial_A\}$ is functional in IDT.

Proof. Follows immediately from Lemma 5.2.1. □

Proposition 5.2.2. Let $A \subseteq AT$. If A is a generalized decision reduct of IDT, then $A \rightarrow \{\partial_{AT}\}$ is a functional dependency in IDT.

Proof. Let A be a generalized decision reduct of IDT. Then $\forall_{x \in O}\ \partial_A(x) = \partial_{AT}(x)$ and, by Proposition 5.2.1, $A \rightarrow \{\partial_A\}$ is a functional dependency in IDT. Hence, $A \rightarrow \{\partial_{AT}\}$ is a functional dependency in IDT. □

According to Proposition 5.2.2, we observe in IDT from Table 3 that $AT \rightarrow \{\partial_{AT}\}$ and $\{b\} \rightarrow \{\partial_{\{b\}}\}$ are functional dependencies. In addition, we observe that $\{b\} \rightarrow \{\partial_{AT}\}$ is a minimal functional dependency in IDT. Nevertheless, there are objects in IDT for which the values of $\partial_{\{b\}}$ and ∂_{AT} differ; for example, $\partial_{AT}(1) \neq \partial_{\{b\}}(1)$. Thus, the minimal functional dependency $\{b\} \rightarrow \{\partial_{AT}\}$ does not imply that $\{b\}$ is a generalized decision reduct.

Theorem 5.2.1. Let $A \to AT$. The existence of a minimal functional dependency $A \to \{\partial_{AT}\}$ in IDT does not imply that A is a generalized decision reduct of IDT.

Corollary 5.2.1. Let $A \subseteq AT$. If A is a generalized decision reduct of IDT, then $A \to \{\partial_{AT}\}$ is a functional dependency, but not necessarily minimal.

Proof. By Proposition 5.2.2 and Theorem 5.2.1. □

5.3 μ-Decision Reducts and Functional Dependencies Under Incompleteness

Lemma 5.3.1. Let $A \subseteq AT$ and $x \in O$. $\forall_{y \in I_A(x)}\ \mu_d^A(y) = \mu_d^A(x)$.

Proof. Analogous to Proof of Lemma 5.2.1; follows from Property 5.1.1b. □

Proposition 5.3.1. Let $A \subseteq AT$. $A \to \{\mu_d^A\}$ is functional in IDT.

Proof. Follows immediately from Lemma 5.3.1. □

Proposition 5.3.2. Let $A \subseteq AT$. If A is a μ-decision reduct of IDT, then $A \to \{\mu_d^{AT}\}$ is a functional dependency in IDT.

Proof. Analogous to the proof of Proposition 5.2.2; follows from the definition of a μ-decision reduct and Proposition 5.3.1. □

Now, we note that $\{b\} \to \{\mu_d^{AT}\}$ is a minimal functional dependency in IDT from Table 3 and $\mu_d^{AT}(1) \neq \mu_d^{\{b\}}(1)$. Thus, the minimal functional dependency $\{b\} \to \{\mu_d^{AT}\}$ does not imply that $\{b\}$ is a μ-decision reduct. Thus, we conclude:

Theorem 5.3.1. Let $A \subseteq AT$. The existence of a minimal functional dependency $A \to \{\mu_d^{AT}\}$ in IDT does not imply that A is a μ-decision reduct of IDT.

Corollary 5.3.1. Let $A \subseteq AT$. If A is a μ-decision reduct of IDT, then $A \to \{\mu_d^{AT}\}$ is a functional dependency, but not necessarily minimal.

Proof. By Proposition 5.3.2 and Theorem 5.3.1. □

5.4 Certain Decision Reducts and Functional Dependencies Under Incompleteness

Lemma 5.4.1. $POS_A = \{x \in O \mid S_A(x) \subseteq I_d(x)\}$.

Proof. $POS_A = \bigcup_{d_i \in V_d} \underline{A}X_{d_i} = \bigcup_{y \in O} \underline{A}I_d(y) = \bigcup_{y \in O} \{x \in O \mid S_A(x) \subseteq I_d(y)\} = /^*$ by Proposition 2.2.1 $^*/ = \bigcup_{y \in O} \{x \in O \mid S_A(x) \subseteq I_d(x)\} = \{x \in O \mid S_A(x) \subseteq I_d(x)\}$. □

Lemma 5.4.2. Let $A \subseteq AT$ and $x \in O$. $\forall_{y \in I_A(x)}\ d_A^{\mathrm{N}}(y) = d_A^{\mathrm{N}}(x)$.

Proof. We shall consider two cases: 1) $x \in POS_A$, and 2) $x \notin POS_A$.

Case 1: By definition, $d_A^{\mathrm{N}}(x) = d(x)$ $(^*)$. By Lemma 5.4.1, $S_A(x) \subseteq I_d(x)$. Hence, and by Property 5.1.1a,b, $\forall_{y \in I_A(x)}\ I_A(y) \subseteq S_A(y) \subseteq I_d(x)$, so, $\forall_{y \in I_A(x)}\ I_d(y) =$

$I_d(x)$ (**). Thus, $\forall_{y \in I_A(x)}\ S_A(y) \subseteq I_d(y)$. Hence, and by Lemma 5.4.1, $\forall_{y \in I_A(x)}$ $y \in POS_A$ (***). By (*), (**) and (**), $\forall_{y \in I_A(x)}\ d_A^{\mathrm{N}}(y) = d(y) = d(x) = d_A^{\mathrm{N}}(x)$.

Case 2: By definition, $d_A^{\mathrm{N}}(x) = \mathrm{N}$ (*), and by Lemma 5.4.1, $S_A(x) \not\subseteq I_d(x)$. Thus, by Proposition 2.2.1, $\neg(\exists_{z \in O}\ S_A(x) \subseteq I_d(z))$. Hence, and by Property 5.1.1b, $\forall_{y \in I_A(x)} \neg(\exists_{z \in O}\ S_A(y) \subseteq I_d(z))$. So, by Proposition 2.2.1, $\forall_{y \in I_A(x)}\ S_A(y) \not\subseteq I_d(y)$. Therefore, $\forall_{y \in I_A(x)}\ y \notin POS_A$. Hence, $\forall_{y \in I_A(x)}\ d_A^{\mathrm{N}}(y) = \mathrm{N} =$ /* by (*) */ $= d_A^{\mathrm{N}}(x)$. □

Proposition 5.4.1. Let $A \subseteq AT$. $A \to \{d_A^{\mathrm{N}}\}$ is a functional dependency in IDT.

Proof. Follows immediately from Lemma 5.4.2. □

Proposition 5.4.2. Let $A \subseteq AT$. If A is a certain decision reduct of IDT, then $A \to \{d_{AT}^{\mathrm{N}}\}$ is a functional dependency in IDT.

Proof. Let A be a certain decision reduct. By the definitions of a certain decision reduct and d_{AT}^{N}, $\forall_{x \in POS}\ S_A(x) \subseteq I_d(x)$ and $d_{AT}^{\mathrm{N}}(x) = d(x)$. Thus, by Property 5.1.1a, $\forall_{x \in POS}\ I_A(x) \subseteq I_{d_{AT}^{\mathrm{N}}}(x)$ (*). By Lemma 5.4.1, $\forall_{x \notin POS}\ S_{AT}(x) \not\subseteq I_d(x)$. Hence, and by Property 5.1.1c, $\forall_{x \notin POS}\ S_A(x) \not\subseteq I_d(x)$. Thus, by Lemma 5.4.1, $\forall_{x \notin POS}\ x \notin POS_A$. Therefore and by the definitions of d_{AT}^{N} and d_A^{N}, $\forall_{x \notin POS}\ d_{AT}^{\mathrm{N}}(x) = \mathrm{N} = d_A^{\mathrm{N}}(x)$. Hence, and by Lemma 5.4.2, $\forall_{x \notin POS}\ \forall_{y \in I_A(x)}$ $d_A^{\mathrm{N}}(y) = d_A^{\mathrm{N}}(x) = \mathrm{N} = d_{AT}^{\mathrm{N}}(x)$. Thus, $\forall_{x \notin POS}\ I_A(x) \subseteq I_{d_{AT}^{\mathrm{N}}}(x)$ (**). By (*) and (**), $\forall_{x \in O}\ I_A(x) \subseteq I_{d_{AT}^{\mathrm{N}}}(x)$. Hence, $A \to \{d_{AT}^{\mathrm{N}}\}$ is a functional dependency. □

Eventually, we note that $\{b\} \to \{d_{AT}^{\mathrm{N}}\}$ is a minimal functional dependency and $d_{AT}^{\mathrm{N}}(1) \neq d_{\{b\}}^{\mathrm{N}}(1)$. Hence, the minimal functional dependency $\{b\} \to \{d_{AT}^{\mathrm{N}}\}$ does not imply that $\{b\}$ is a certain decision reduct. Thus we conclude:

Theorem 5.4.1. Let $A \subseteq AT$. The existence of a minimal functional dependency $A \to \{d_{AT}^{\mathrm{N}}\}$ in IDT does not imply that A is a certain decision reduct of IDT.

Corollary 5.4.1. Let $A \subseteq AT$. If A is a certain decision reduct of IDT, then $A \to \{d_{AT}^{\mathrm{N}}\}$ is a functional dependency, but not necessarily minimal.

Proof. By Proposition 5.4.2 and Theorem 5.4.1. □

6 Conclusions

Certain decision reducts, generalized decision reducts, and membership distribution reducts are provable to be sets of conditional attributes that functionally determine respective modifications of a decision attribute both in complete and incomplete information systems. We have also proved, however, that, unlike in the case of complete systems, the reducts in incomplete systems are not guaranteed to be minimal sets of conditional attributes that functionally determine respective modifications of the decision attribute.

References

1. Järvinen, J.: Pawlak's Information Systems in Terms of Galois Connections and Functional Dependencies. Fundamenta Informaticae 75, 315–330 (2007)
2. Kryszkiewicz, M.: Rough Set Approach to Incomplete Information Systems. Journal of Information Sciences 112, 39–49 (1998)
3. Kryszkiewicz, M.: Properties of Incomplete Information Systems in the Framework of Rough Sets. In: Studies in Fuzziness and Soft Computing 18. Rough Sets in Knowledge Discovery 1, pp. 442–450. Physica Verlag, Heidelberg (1998)
4. Kryszkiewicz, M.: Rules in Incomplete Information Systems. Journal of Information Sciences 113, 271–292 (1999)
5. Kryszkiewicz, M.: Rough Set Approach to Rules Generation from Incomplete Information Systems. The Encyclopedia of Computer Science and Technology. Marcel Dekker, Inc. New York, vol. 44, pp. 319–346 (2001)
6. Kryszkiewicz, M.: Comparative Study of Alternative Types of Knowledge Reduction in Inconsistent Systems. Int'l Journal of Int. Systems 16(1), 105–120 (2001)
7. Kryszkiewicz, M.: Comparative Study of Alternative Types of Knowledge Reduction in Inconsistent Systems - Revised. ICS Research Report 13/2004, Warsaw (October 2004)
8. Lin, T.Y.: An Overview of Rough Set Theory from the Point View of Relational Databases. Bulletin of International Rough Set. Society 1(1), 30–34 (1998)
9. Nguyen, H.S.: Approximate Boolean Reasoning: Foundations and Applications in Data Mining. In: Peters, J.F., Skowron, A. (eds.) Transactions on Rough Sets V. LNCS, vol. 4100, pp. 334–506. Springer, Heidelberg (2006)
10. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data, vol. 9. Kluwer Academic Publishers, Boston (1991)
11. Pawlak, Z., Skowron, A.: Rudiments of Rough Sets, Inf. Sci. 177(1) 3–27
12. Pawlak, Z., Skowron, A.: Rough Sets and Boolean Reasoning, Inf. Sci. 177(1) 41–73
13. Skowron, A.: Boolean Reasoning for Decision Rules Generation. ISMIS, 295–305 (1993)
14. Skowron, A., Rauszer, C.: The Discernibility Matrices and Functions in Information Systems. In: Intelligent Decision Support. Handbook of Applications and Advances of Rough Sets Theory, pp. 331–362. Kluwer, Dordrecht (1992)
15. Slezak, D.: Approximate Reducts in Decision Tables. IPMU 3, 1159–1164 (1996)
16. Slezak, D.: Searching for Frequential Reducts in Decision Tables with Uncertain Objects. RSCTC, 52–59 (1998)
17. Slezak, D.: Approximate Entropy Reducts. Fundam. Inform. 53(3-4), 365–390 (2002)
18. Slezak, D.: Association Reducts: Complexity and Heuristics. RSCTC, 157–64 (2006)
19. Slezak, D.: Association Reducts: Boolean Representation. RSKT, 305–312 (2006)