Fundamenta Informaticae 67 (2005) 131–147
IOS Press

Flexible Indiscernibility Relations for Missing Attribute Values

Rafał Latkowski
Warsaw University, Institute of Computer Science,
ul. Banacha 2, 02-097 Warszawa, Poland,
R.Latkowski@mimuw.edu.pl

Abstract. The indiscernibility relation is a fundamental concept of the rough set theory. The original definition of the indiscernibility relation does not capture the situation where some of the attribute values are missing. This paper tries to enhance former works by proposing an individual treatment of missing values at the attribute or value level. The main assumption of the theses presented in this paper considers that not all missing values are semantically equal. We propose two different approaches to create an individual indiscernibility relation for a particular information system. The first relation assumes variable, but fixed semantics of missing attribute values in different columns. The second relation assumes different semantics of missing attribute values, although this variability is limited with expressive power of formulas utilizing descriptors. We provide also a comparison of flexible indiscernibility relations and missing value imputation methods. Finally we present a simple algorithm for inducing sub-optimal relations from data.

Keywords: Rough Sets, Missing Attribute Values, Incomplete Information Systems

1. Introduction

The concept of similarity or discernibility is important in many kinds of reasoning. It is very essential not only for the rough set theory, but also for all other aspects of reasoning. Its importance arises from the fact, that almost every other concept utilized in reasoning and especially in machine learning depends on the similarity or discernibility. For example, if the reasoning process is carried out using a set of objects, then it is necessary to know which objects are basically the same or describe the same situation. The other example is the decision rule matching. Before applying a decision rule, it has to be compared to an object, in order to determine does the object is somehow similar enough to the decision rule. Also a decision rule should be identically applicable to objects that are indiscernible. The semantics of the indiscernibility relation impacts on soundness of reasoning.

The indiscernibility relation is a fundamental concept of the rough set theory. The original definition of the indiscernibility relation, thus the rough set theory, does not capture the situation where some of

the attribute values are missing. The problem of missing values handling within the rough set framework has been already faced in literature, e.g., by Grzymała [8, 10], Słowiński [29], Stefanowski [31] and Kryszkiewicz [13]. The proposed approaches consider alternative definitions of the indiscernibility relation, which reflect various semantics of missing attribute values. The main difficulty of applied alternatives for the indiscernibility relation arises from semantics fixed in advance of all missing values in whole information system, what was identified in, e.g., [31]. This paper tries to enhance former works by proposing an individual treatment of missing values at the attribute or value level.

The main assumption of the theses presented in this paper considers that not all missing values are semantically equal. Among a number of taxonomies (see, e.g., [3, 18, 31]) for missing attribute value semantics, the two main types of missing values can be determined: the existential null as an unknown value of considered property, called also "missing" semantics and the placeholder null as an inapplicable value for considered property, what is similar to the "absent" missing value semantics. These two main types of missing attribute values possibly can be even mixed together in one database column, in a way precluding the distinguishing of one type from another. The different meanings of missing attribute values obviously have an impact on the concept of the indiscernibility relation and in consequence on the concept of certain and approximate decision rules. We expect the decision rules induced with help of an indiscernibility relation customized to a particular decision system to perform better in terms of knowledge discovery and classification accuracy.

In this paper we propose two different approaches to create an individual indiscernibility relation for a particular information system. The first approach — attribute constrained indiscernibility relation — assumes variable, but fixed semantics of missing attribute values in different columns. In other words treatment of missing values by attribute constrained indiscernibility relation can be different for different attributes (columns), but all missing values from one attribute have to be handled in exactly the same way. The second approach — descriptor constrained indiscernibility relation — assumes different semantics of missing attribute values, although this variability is limited with expressive power of formulas utilizing descriptors. It means that treatment of missing attribute values by descriptor constrained indiscernibility relation can be differentiated using logic formulas over descriptor language for a given information system.

The paper is organized as follows. The next section describes classic definition of the indiscernibility relation together with standard approaches to handle missing attribute values. In the third section the idea of flexible indiscernibility relations is presented. The fourth section describes connections between flexible indiscernibility relations and missing value imputation methods. In the fifth section an algorithm for inducing optimal attribute-constrained indiscernibility relations is presented together with empirical evaluation. The last, sixth section contains final conclusions.

2. Preliminaries

The indiscernibility relation is formulated on objects belonging to an *information system* (see, e.g., [11, 22]).

Definition 2.1. An information system $\mathbb{A} = (U, A)$ is a pair, where U is a non-empty, finite set of objects called the universe and A is a non-empty, finite set of attributes. Attributes $a \in A$ are functions $a : U \to V_a$, where V_a is a domain of attribute a.

In its original formulation an information system concerns a situation where attribute values for all objects are known. Such an information system is called *complete information system*. The classic indiscernibility relation is formulated for complete information systems (cf. [11, 22]).

Definition 2.2. Let $\mathbb{A} = (U, A)$ be a (complete) information system and $B \subseteq A$. The indiscernibility relation IND_B is defined as follows:

$$\mathrm{IND}_B = \{(x, y) \in U \times U : \forall_{a \in B}\, a(x) = a(y)\}. \tag{1}$$

The classic indiscernibility relation is an equivalence relation, i.e., is reflexive, symmetric and transitive. The equivalence classes of the indiscernibility relation form upper and lower concept approximations (see, e.g., [11, 22]), which are used in inductive learning to find certain and plausible knowledge.

In a presence of missing data the definition of information system has to be extended. There are several ways of defining an incomplete information system, which include extending attribute domains, applying partial functions as attributes and others. Also missing values are differently notated using one or more special symbols (cf. [9]), or even without using any special symbol as in the case of partial functions. In this paper we will use extended domain of attribute values, while such an approach is closest to the real-life implementations of incomplete information systems and delivers easy to use notation.

Definition 2.3. An incomplete information system $\mathbb{A} = (U, A)$ is a pair, where U is a non-empty, finite set of objects called the universe and A is a non-empty, finite set of attributes. Attributes $a_i \in A$ are functions $a : U \to V_a^*$, where V_a is a domain of attribute a, $V_a^* = V_a \cup \{*\}$ and $* \notin V_a$.

The special symbol "$*$" denotes absence of regular attribute value and if $a(x) = *$ we say that a is not defined on x. In the relational databases there exists a similar notion — "NULL" that represents missing attribute value in a database record (see, e.g., [4, 18]). In above definition we are using only one symbol for missing attribute value for several reasons. Firstly, in all systems for data gathering and processing missing values are stored using only one symbol. Even in systems that have a possibility to represent several types of missing values, only one symbol is practically used. Secondly, even if an information system contains missing values of several types, but there is no additional information that allows differentiating them, then it is a task for machine learning methods to detect those types and apply to them proper semantics and treatment.

There is no necessity to use different notation for complete and incomplete information systems. If all attribute values are defined and there is no necessity to use special symbol for denoting missing attribute value, then both definitions are compatible. Moreover, we can consider also an extension of indiscernibility relation that works on incomplete information systems. The above definition of the indiscernibility relation IND is not specified on missing attribute values. Many researchers proposed different approaches to extend this relation, some of them are described later. One extension of the indiscernibility relation is the most similar and compatible with the classic definition. If we consider the standard equality relation on any domain, than equal are only two exactly the same elements from the domain. Similarly, for any domain V_a^* the standard equality relation holds for $* = *$ and for all values other than $*$ equality $v = *$ and $* = v$ does not hold. Using such property of equality relation the above definition of the indiscernibility relation IND is applicable also to incomplete information systems. Moreover, this extension inherits all properties of the classic indiscernibility relation IND and also, due to the way of representing missing attribute values in computers, any implementation of the

classic indiscernibility relation will behave like exactly this extension of the indiscernibility relation, when applied to incomplete data. In rest of this paper we will always assume this extension of equality relation.

In such a case the indiscernibility relation IND is still an equivalence relation and two objects are indiscernible if and only if their descriptions on considered attributes are identical. Such treatment of missing attribute values corresponds to a placeholder-null semantics which assumes that a missing attribute value is a consequence of the fact, that an object cannot be described by any value on considered attribute. In contrast to further considered semantics, such a missing value is meaningful in reasoning or, at least, is not comparable to any domain value.

It is known fact, that missing value handling by IND relation can decrease the correctness of inductive reasoning (see, e.g., [32]). If for example a medical examination was not carried out on two patients, we cannot suppose that they are similar. To overcome this problem some other indiscernibility relations were proposed for an alternative missing values handling within the rough set framework. However, none of them is universally the best nor always correct in the terms of correctness of reasoning and size of upper and lower approximations (cf. [31, 32, 10]). To overcome this problem also some other approaches were proposed, where the additional numerical tuning of the indiscernibility relation is made or fuzzy sets framework is utilized (see, e.g., [31, 32, 7]). The common problem of all these numerical approaches is lack of algorithm that selects optimal parameters and shapes of fuzzy membership functions for a considered information system. The two most important indiscernibility relation replacements are *symmetrical similarity* relation and *unsymmetrical similarity* relation.

Definition 2.4. Let $\mathbb{A} = (U, A)$ be an incomplete information system and $B \subseteq A$. The symmetrical similarity relation SS_B is defined as follows:

$$\mathrm{SS}_B = \{(x, y) \in U \times U : \forall_{a \in B}\ a(x) = a(y) \vee a(x) = * \vee a(y) = *\}. \tag{2}$$

The symmetric similarity relation, called also tolerance relation was inspired by week equalities for partial structures in universal algebra and was frequently considered within the rough set framework (see, e.g., [13, 32]). Its interpretation is related with missing-null semantics, where missing values are lost or not stored for a variety of reasons, but at least theoretically it is possible to assign to an object a value for considered attribute. The symmetrical similarity relation is reflexive, symmetric but not necessarily transitive. It does not form equivalence classes, so the definition of upper and lower concept approximation has to be modified (see, e.g., [8, 32] for details).

Definition 2.5. Let $\mathbb{A} = (U, A)$ be an incomplete information system and $B \subseteq A$. The unsymmetrical similarity relation US_B is defined as follows:

$$\mathrm{US}_B = \{(x, y) \in U \times U : \forall_{a \in B}\ a(x) = a(y) \vee a(x) = *\}. \tag{3}$$

The unsymmetrical similarity relation considers another interpretation of missing values, where it is not possible to describe an object with any value of considered attribute (see, e.g., [32]). In contrast to the classic indiscernibility relation case, the missing attribute value is unimportant for reasoning process. An original-copy example is frequently mentioned as a justification of unsymmetrical similarity, where it is quite natural to call a copy to be similar to the original, but it is very odd to call the original to be similar to a copy. The unsymmetrical similarity relation is reflexive, transitive but not necessarily symmetric, so

also upper and lower concept approximation have to be redefined. There exists also a general definition of upper and lower concept approximation that works in the case of symmetrical and unsymmetrical similarity and also classic indiscernibility relation.

Three introduced indiscernibility relations can be characterized by following theorem formulated by Stefanowski in [31].

Theorem 2.1. For any incomplete information system $\mathbb{A} = (U, A)$ and subset of attributes $B \subseteq A$ following property holds:

$$\mathrm{IND}_B \subseteq \mathrm{US}_B \subseteq \mathrm{SS}_B. \tag{4}$$

The above theorem is very important for all possible approaches to indiscernibility relation. In this paper we will not discuss it in detail, but it is very reasonable to take two following assumptions for every indiscernibility relation. Firstly, the indiscernibility relation should be reflexive. Such an assumption is very common and even some definitions or implementations of information systems do not consider contrary situation. Moreover, it would rise many problems if we do not assume that an object is similar to itself. Secondly, the indiscernibility relation should not join objects with different defined values on the same attribute. Joining of completely different objects can follow from utilizing additional knowledge, which should be represented elsewhere, either in data preparation or in decision rule induction. There exist another stream of research in rough set theory, namely tolerance relations and tolerance spaces (see, e.g., [12, 21, 24, 25]), where such assumption is not made. However, this research is rather related with feature extraction, but not with missing value handling. If we make two above assumptions, then the IND_B is the smallest indiscernibility relation, in inclusion sense, that satisfies assumption on reflexivity, while SS_B is the biggest indiscernibility relation, in inclusion sense, that satisfies second assumption.

3. Flexible Indiscernibility Relations

The main difficulty of applied alternatives for classic indiscernibility relation arises from the fact that semantics of all missing values is fixed in whole information system. It has been observed (see, e.g., [31]), that presented above indiscernibility relations have some deficiencies in creating relevant and big enough upper and lower approximations of considered concept.

In this paper we try to find another way to provide a flexible indiscernibility relation by using logical approach. It means that the indiscernibility relation should be expressed as a logical formula without any additional numeric parameters. This gives us the possibility to apply boolean reasoning methods that proved its usefulness many times (see, e.g., [26]). We believe that indiscernibility relation based on logical formula would be easier for automatic generation or induction using boolean reasoning.

We propose two different approaches to create an individual indiscernibility relation for a particular information system. The first relation assumes variable, but fixed semantics of missing attribute values in different columns. The second relation assumes different semantics of missing attribute values, although this variability is limited with expressive power of formulas utilizing descriptors.

3.1. Attribute Constrained Indiscernibility Relation

The attribute constrained indiscernibility relation allows utilizing different missing value semantics for each attribute.

Definition 3.1. An attribute indiscernibility relation $\mathrm{I}_a \subseteq V_a^* \times V_a^*$ is a reflexive relation that for any pair in relation $(v_1, v_2) \in \mathrm{I}_a$ satisfies condition that if both values are non-missing, then they are equal, i.e., $((v_1, v_2) \in \mathrm{I}_a) \Rightarrow ((v_1 \neq * \wedge v_2 \neq *) \Rightarrow (v_1 = v_2))$.

The above implication can be rewritten also as $((v_1, v_2) \in \mathrm{I}_a) \Rightarrow (v_1 = v_2 \vee v_1 = * \vee v_2 = *)$. It guarantees full compatibility with standard equality on defined attribute values, but leave freedom in treatment of missing values. Relation I_a can be interpreted as a generalization of equality, which is reflexive, symmetric at least where standard equality is symmetric, and not necessarily transitive. Moreover, if we identify a one-attribute information system $\mathbb{A} = (U, \{a\})$ with attribute a for which attribute indiscernibility relation I_a is constructed, then this relation can be also interpreted as an indiscernibility relation for such one-attribute information system $\mathbb{A} = (U, \{a\})$. Let us give some examples of attribute indiscernibility relations:

- $\{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2\}$,
- $\{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2 \vee v_1 = *\}$,
- $\{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2\} \cup \{(3.14, *), (*, 3.14)\}$, assuming that $3.14 \in V_a$.

Definition 3.2. Let $\mathbb{A} = (U, A)$ be an incomplete information system and $B \subseteq A$ be a subset of attributes such that attributes $a \in B$ are functions $a : U \to V_a^*$. An indiscernibility relation is attribute constrained indiscernibility relation if it can be represented in following form:

$$\mathrm{AL}_B = \{(x, y) \in U \times U : \forall_{a \in B}(a(x), a(y)) \in \mathrm{I}_a\}, \tag{5}$$

for some attribute indiscernibility relations I_a.

We may say by analogy to linear independence, that the attribute constrained indiscernibility relation is any attribute independent relation constructed from such attribute indiscernibilities. Such a construction allows to obtain different semantics of missing attribute values for each attribute. The name of the relation suggests, that flexibility in using different semantics of missing attribute values is limited by fixing it for an attribute.

To better explain the application area of such a relation let take an example of information system $\mathbb{A} = (U, \{c, w, p, ec\})$, containing descriptions of motorcycles and bicycles. Motorcycles and bicycles both have color (c), weight (w) and price (p). However, the engine capacity (ec) is a property, which does not make any sense in case of bicycles. In such an example the missing values in color, weight and price can be treated using missing-null semantics, while missing values in engine capacity can be treated as placeholder-null semantics. The simplest attribute constrained indiscernibility relation formula representing the above example can be:

$$\begin{aligned} \mathrm{AL}_{\{c,w,p,ec\}}(x, y) \quad &= \quad (c(x) = c(y) \vee c(x) = * \vee c(y) = *) \\ &\wedge \quad (w(x) = w(y) \vee w(x) = * \vee w(y) = *) \\ &\wedge \quad (p(x) = p(y) \vee p(x) = * \vee p(y) = *) \\ &\wedge \quad ec(x) = ec(y). \end{aligned} \tag{6}$$

The relation $\mathrm{AL}_{\mathbb{A}}$ implements for attributes c,w and p the existential missing value semantics, while for attribute ec the placeholder missing value semantic.

Let take a closer look at the building blocks of attribute constrained indiscernibility relations — attribute indiscernibility relations.

Theorem 3.1. Every attribute indiscernibility relation $\mathrm{I}_a : V_a^* \times V_a^*$ satisfies following property:

$$\{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2\} \subseteq \mathrm{I}_a \subseteq \{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2 \vee v_1 = * \vee * = v_2\} \quad (7)$$

Proof:
The first inequality results directly from the reflexivity assumption of Definition 3.1. Every attribute indiscernibility should be reflexive, so every pair $(v, v) \in \mathrm{I}_a$ and first inequality is satisfied. This inequality is not necessarily strict, because the relation on the left satisfies all assumptions of attribute indiscernibility relation. The second inequality results from the condition in Definition 3.1. Let assume that a pair (v_1, v_2) is an element of attribute indiscernibility relation. If $v_1 = *$ is a missing value or $v_2 = *$ is a missing value, than such pair is an element of the right relation. In other case both v_1 and v_2 are defined and they have to be equal from definition of attribute indiscernibility relation, so such pair is an element of the right relation. The right inequality is also not necessarily strict, while the right relation satisfies reflexivity assumption and the condition from Definition 3.1. □

The above theorem provides the upper and lower limit for attribute indiscernibilities. For attribute constrained indiscernibility relations similar property holds. In fact IND_B and SS_B relations are also attribute constrained indiscernibility relations. However, an arbitrary attribute constrained indiscernibility relation is not comparable with unsymmetrical similarity relation US_B.

Theorem 3.2. The classic indiscernibility relation IND_B and symmetrical similarity relation SS_B are respectively the smallest and the biggest attribute constrained indiscernibility relation, i.e., following property holds for any information system $\mathbb{A} = (U, A)$, $B \subseteq A$ and attribute constrained indiscernibility relation AL_B:

$$\mathrm{IND}_B \subseteq \mathrm{AL}_B \subseteq \mathrm{SS}_B \quad (8)$$

Proof:
This theorem results from Theorem 3.1 and Definition 3.2. Firstly, let us observe that if we construct an attribute constrained indiscernibility relation using attribute indiscernibility relations $\mathrm{I}_a^{\mathrm{IND}} = \{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2\}$ for each attribute $a \in B$, then we will get the classic indiscernibility relation IND_B. Similarly, if we construct an attribute constrained indiscernibility relation using attribute indiscernibility relations $\mathrm{I}_a^{\mathrm{SS}} = \{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2 \vee v_1 = * \vee v_2 = *\}$ for each attribute $a \in B$, then we will get the symmetrical similarity relation SS_B. Secondly, every attribute constrained indiscernibility relation AL_B is constructed by conjunction of attribute indiscernibilities I_a for each attribute $a \in B$. If we combine separate inequalities from Theorem 3.1 and merge them with conjunctions according to the Definition 3.2, then we get inequalities as in the above theorem. □

We can consider also a special subfamilies of attribute constrained indiscernibility relations. For example let assume, that the building blocks, attribute indiscernibility relations are constrained to be only one of these three possibilities:

- $\mathrm{I}_a^{\mathrm{IND}} = \{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2\}$,

- $\mathrm{I}_a^{\mathrm{US}} = \{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2 \vee v_1 = *\}$,
- $\mathrm{I}_a^{\mathrm{SS}} = \{(v_1, v_2) \in V_a^* \times V_a^* : v_1 = v_2 \vee v_1 = * \vee * = v_2\}$.

These three attribute relations generate smaller family of attribute constrained indiscernibility relations, where on each attribute only one of three previously specified missing value semantics can be used. This subfamily contains IND, US and SS relations. Moreover any relation from this family can be labelled by set of attribute indiscernibility relations used in its construction. For example, if $B = \{a_1, a_2, \ldots, a_n\}$, then IND_B relation can be labelled by $\{\mathrm{I}_{a_1}^{\mathrm{IND}}, \mathrm{I}_{a_2}^{\mathrm{IND}}, \ldots, \mathrm{I}_{a_n}^{\mathrm{IND}}\}$, US_B relation can be labelled by $\{\mathrm{I}_{a_1}^{\mathrm{US}}, \mathrm{I}_{a_2}^{\mathrm{US}}, \ldots, \mathrm{I}_{a_n}^{\mathrm{US}}\}$ and finally SS_B relation can be labelled by $\{\mathrm{I}_{a_1}^{\mathrm{SS}}, \mathrm{I}_{a_2}^{\mathrm{SS}}, \ldots, \mathrm{I}_{a_n}^{\mathrm{SS}}\}$. These labels simplify search space of possible attribute constrained indiscernibility relations. Using such labels both systematic search methods and other search methods such as genetic algorithms can efficiently search for optimal attribute constrained indiscernibility relation using specified optimization criterion. In fact this search space is very regular and form a lattice with the smallest element IND_B, the biggest element SS_B and 3^n elements in lattice. This lattice is similar to the lattice of subsets of set of attributes B that contains 2^n elements, but here each attribute can have three states (i.e., $\mathrm{I}^{\mathrm{IND}}$, I^{US} and I^{SS}), while in lattice of subsets each attribute can be only present or absent. Moreover a general family of attribute constrained indiscernibility relations also create a lattice, but structure of this lattice is not as regular as previous one.

3.2. Descriptor Constrained Indiscernibility Relation

The descriptor constrained indiscernibility relation gives more flexibility than attribute constrained indiscernibility relation. In this case the relation in not limited to fixed missing value semantics for an attribute, but the relation can be described with any propositional logic formula over descriptors from information system.

Definition 3.3. Let $\mathbb{A} = (U, A)$ be an incomplete information system. A formula of language $\mathcal{DL}_{\mathbb{A}}$ can be one of the following:

- $a(x) = v$ — descriptor-value equality,
- $a(x) = *$ — descriptor-missing value equality,
- $a(x) = b(y)$ — descriptor-descriptor equality,
- $\neg\phi$ — negation of formula $\phi \in \mathcal{DL}_{\mathbb{A}}$,
- $\phi \vee \psi$ — alternative of formulas $\phi \in \mathcal{DL}_{\mathbb{A}}$ and $\psi \in \mathcal{DL}_{\mathbb{A}}$,
- $\phi \wedge \psi$ — conjunction of formulas $\phi \in \mathcal{DL}_{\mathbb{A}}$ and $\psi \in \mathcal{DL}_{\mathbb{A}}$.

An indiscernibility relation is descriptor constrained indiscernibility relation if it can be represented by a formula $\phi \in \mathcal{DL}_{\mathbb{A}}$.

Also a simpler language for defining descriptor-based formulas can be considered, where only descriptor-descriptor equalities on the same attribute are allowed (i.e., $a(x) = a(y)$), and without negation. However, the expressive power of such language is exactly the same, because we are considering only finite

universes and finite attribute sets. The more advanced expressions can be transcribed using other descriptors and in worst case by enumeration of cases, what usually yields in dramatic growth of expression length.

Continuing the above example, let's consider information system $\mathbb{A} = (U, \{c, w, p, ec, t\})$, where $t : u \rightarrow \{b, m\}$ is an attribute describing type of object: for motorcycle $t(u) = m$ and for bicycle $t(u) = b$. Let also assume for simplicity that type attribute t does not contain missing values. If we assume, that the values of engine capacity can be also missing in case of motorcycles, what should be treated as existential null rather than placeholder null, than the simple descriptor constrained indiscernibility relation formula can be as follows:

$$\begin{aligned} \mathrm{DL}_{\mathbb{A}}(x,y) &= (c(x) = c(y) \vee c(x) = * \vee c(y) = *) \\ &\wedge (w(x) = w(y) \vee w(x) = * \vee w(y) = *) \\ &\wedge (p(x) = p(y) \vee p(x) = * \vee p(y) = *) \\ &\wedge (ec(x) = ec(y) \vee (t(x) = m \wedge t(y) = m \wedge (ec(x) = * \vee ec(y) = *))) \\ &\wedge (t(x) = t(y)). \end{aligned} \tag{9}$$

The relation $\mathrm{DL}_{\mathbb{A}}$ implements for attributes c,w and p the existential missing value semantics as well as for the attribute ec, when both objects are motorcycles. If one or two of the considered objects are bicycles, then relation $\mathrm{DL}_{\mathbb{A}}$ implements for attribute ec the placeholder missing value semantic.

Using assumptions presented in second section, which state that for any indiscernibility relation IND_B and SS_B are respectively the smallest and the biggest indiscernibility relation, we can easily observe that family of descriptor constrained indiscernibility relations should form a lattice. However, in this case structure of this family remains unknown. The complexity of this structure and the number of possible descriptor constrained indiscernibility relations suggest, that for selecting (sub)optimal in some sense relation perhaps not the search algorithm should be used, but rather inductive learning or boolean reasoning.

3.3. Free Indiscernibility Relation

There exists the possibility to create a free indiscernibility relation which would not be limited to the attribute nor descriptor expressive power. Such a relation gives an opportunity to capture all possible relationships between objects considered in information system and semantics of missing attribute values. However, exceeding the limits of expressive power of propositional formulae language over descriptors precludes usability of such a relation. Without the description of indiscernibility relation formulated in language easily decidable we are not able to apply such relation correctly to new, unseen objects. If the relation does not contain any (decidable) description, then the only way to characterize it is the enumeration of elements in relation, e.g., in form of relation matrix. If the matrix does not contain unseen objects, then we are not able to determine whether the particular new object is in the relation with any other or is not. This suggest that such approach is useless, unless we try to do some approximation of such relation using attribute- or descriptor constrained indiscernibility relation.

4. Imputation

Within machine learning and especially statistical approach to machine learning the imputation methods are frequently considered way of handling missing attribute values (see, e.g., [5, 6, 10, 19]. Generally speaking imputation methods cope with missing attribute values by replacing them by regular attribute values or a set of acceptable values. Imputation methods can not substitute algorithms for inducing optimal indiscernibility relation, but it is interesting to compare their expressive power and present a natural transformation of imputations into indiscernibility relations.

The simplest methods for missing value imputation are imputations using mean value, median, most common value and other statistics or manually chosen values. All these methods at first estimate replacement value for each attribute and then replace all missing attribute values with it. Such way of missing value handling can be easily implemented using an attribute constrained indiscernibility relation. Let assume that values estimated by an imputation algorithm for attributes $A = \{a_1, a_2, \ldots, a_n\}$ are respectively $r_1, r_2, \ldots, r_n$, where $r_i \in V_{a_i}$. We can create attribute indiscernibility relations $I_{a_1}, I_{a_2}, \ldots, I_{a_n}$ defined as $I_{a_i} = \{(v_1, v_2) \in V_{a_i}^* \times V_{a_i}^* : v_1 = v_2\} \cup \{(r_i, *), (*, r_i)\}$ and using them we can construct the attribute constrained indiscernibility relation that handles missing values exactly in the same way as the imputation method does. Similar way of proceeding can be applied to an imputation method that does not consider inter-attribute relations, where missing values are replaced with a set of possible values. Moreover in the case of indiscernibility relation we do not have to extend the definition of an information system to be able to express sets of values in place of attribute values.

Not all imputation algorithms can be expressed using attribute constrained indiscernibility relations. An example of such algorithm is a modification of previously described imputations, where replacement values are estimated not over all objects, but only within clusters (see, e.g., [5]). If only the clusters can be described by descriptors, what is true for presented in [5] NCBMM and RCBMM method, but not for the KMCMM, then these descriptions can be used in conjunction with replaced value in formulation of descriptor constrained indiscernibility relation. In general, descriptor constrained indiscernibility relations can represent any deterministic imputation, where determinism is understand as the fact that every identically described object is imputed with exactly the same value or set of values. In such a case we simply add a conjunction of descriptor that describe considered object to the formula together with clause describing possible replacement of missing attribute values. An example of the method that does not satisfy this condition is the EM imputation method, where every object is imputed with value randomized from a probability distribution. Although there is a way to bypass this limitation by introducing an artificial attribute simulating nondeterministic behavior, we must state that descriptor constrained indiscernibility relations are not designed to cope with nondeterminism.

5. Inducing Optimal Attribute Constrained Indiscernibility Relations

The concept of attribute constrained indiscernibility relation is intentionally designed to be relatively simple. Especially the subfamily of attribute constrained indiscernibility relations that utilizes only three standard indiscernibility relations is designed to be relatively small and well structured. The most important property of this family is the fact, that for an information system with N conditional attributes there are 3^N different attribute constrained indiscernibility relations. Such a reasonable number of possibilities, especially where N is small, makes it possible to employ simple strategies to search for an optimal

indiscernibility relation in some sense.

5.1. Approach Description

Our approach to induction of optimal attribute constrained indiscernibility relations is related with reduction to a search problem. We carried out some experiments in order to test the possibility of selecting the best attribute constrained indiscernibility relation for a particular information system. For this work we simply assumed that *the best* attribute constrained indiscernibility relation is this one that results in the best accuracy of classification. The process of searching for best relation or best accuracy is somehow similar to feature selection problem (see, e.g., [27, 27, 33]) or to learning tolerance or similarity relations (see, e.g., [12, 21, 24]). If we consider all classifiers induced from considered information system using the same (deterministic) method, but different indiscernibility relation, then the classifier that achieves the best classification accuracy determines the best indiscernibility relation. Obviously, such an indiscernibility relation is not necessary uniquely determined. For example, in the case of complete information system (i.e. information system without missing values) all attribute constrained indiscernibility relations are equally good.

In the case of searching for optimal indiscernibility relation the space of possible solutions is well defined. A frequent approach to such problems is to define a special heuristic function that evaluates possible solutions and then to utilize a search algorithm to find a (sub)optimal solution. Following this guidelines it is necessary to create a special heuristi function that approximates quality of classification using selected indiscernibility relation. This is not an easy task, because the induction of decision rules from an information system is a complex process. From the other side such approximation is desirable, since applying full classification algorithm would be very time consuming, especially without advanced search strategy. Apart from the positive region, the most important property that characterizes indiscernibility together with information system is the number of indiscernible object pairs. The number of indiscernible object pairs from different decision classes is frequently used in so called information measures (see, e.g., [20, 27, 28, 33]). One example of such measure is the conflict measure utilized in rough set approach to decision tree induction, where different attributes or potential cuts are compared under constant indiscernibility relation (see, e.g., [20]). In this case a similar measure can be used, but with some modifications that enables comparing different indiscernibility relations. A heuristic function that satisfies this condition is the number of indiscernible objects from the same decision class divided by the number of indiscernible objects at all.

$$h(R) = \frac{card\{(x,y) \in U \times U : (x,y) \in R \wedge d(x) = d(y)\}}{card\{(x,y) \in U \times U : (x,y) \in R\}}, \tag{10}$$

where R is an arbitrary indiscernibility relation defined on information system $\mathbb{A} = (U, A \cup \{d\})$, and d is a decision attribute. This function takes values from 0 to 1, where $h(R) = 0$ means that all indiscernible pairs are in conflict, while $h(R) = 1$ means that there are conflicts at all. This measure is similar to the measure that evaluates normalized size of the positive region (cf. [11, 22]), but different. During the initial experiments it turned out that function $h(R)$ does not differentiate enough various indiscernibility relations. We had to modify the heuristic function used for searching optimal indiscernibility and replace in equation the fact of satisfying relation R by number of attributes that would satisfy R truncated to

those attributes.

$$h'(R) = \frac{\sum_{(x,y)\in U\times U:d(x)=d(y)} card\{a \in A : (x,y) \in R_{|\{a\}}\}}{\sum_{(x,y)\in U\times U} card\{a \in A : (x,y) \in R_{|\{a\}}\}} \tag{11}$$

In other words, in numerator we count for each pair of objects $(x, y) \in U \times U$ that have the same decision $d(x) = d(y)$ the number of attributes on which objects x and y are indiscernible and in denominator we count for each pair of objects $(x, y) \in U \times U$, irrespective of their decisions, exactly the same number of attributes on which objects x and y are indiscernible. Such a function gives more diversified results than function $h(R)$. In experimental part of our work we decided to test the function $h'(R)$, whether it can be used as heuristic function that approximates goodness of an indiscernibility relation R or not.

5.2. Data Sets and Data Preprocessing

Experiments were carried out using Distributed Executor for Rough Set Exploration System (DIXER 2.0.6) that allows executing in grid of workstation experiments that utilize Rough Set Exploration System (RSES 2.1) algorithms (cf. [2]). In our experiments we used following five data sets:

- ech — Echocardiogram data, 2 classes, 5 numerical attributes, 1 categorical attribute, 131 observations, 17.6% incomplete cases, 4.7% missing values.
- hin — Head injury data, 3 classes, 6 categorical attributes, 1000 observations, 40.5% incomplete cases, 9.8% missing values.
- hur2 — Hurricanes data, 2 classes, 6 numerical attributes, 209 observations, 10.5% incomplete cases, 1.8% missing values.
- pid2 — Pima Indians diabetes , 2 classes, 8 numerical attributes, 768 observations, 48.8% incomplete cases, 10.4% missing values.
- smo2 — Attitudes towards workplace smoking restrictions, 3 classes, 4 numerical attributes, 4 categorical attributes, 2855 observations, 18.7% incomplete cases, 2.5% missing values.

All data sets were taken from *Recursive-Partitioning.com* [17]. The selection of these data sets was based on number of conditional attributes (no more than eight), amount of missing attribute values and their documented natural origin. Data sets are originally partitioned in order to use the ten-fold cross-validation (CV10) and all results are obtained using CV10 that utilize this partition.

During empirical evaluation we encounter some obstacles. We have chosen optimal (exhaustive) rule induction algorithm for verification of proposed approach, because it directly depends on definition of indiscernibility relation and is a very standard algorithm developed within rough set framework. Unfortunately, every implementation of this method contain indiscernibility relation hard-coded inside the algorithm. Moreover also other standard algorithms for classifier induction and their implementations contain indiscernibility relation hard-coded, so fixed for execution of algorithm for any information system. Rough Set Exploration System (RSES 2.1) contain implementation of optimal rule induction algorithm. This implementation allows decision rule induction using classic indiscernibility and symmetrical similarity, but not simultaneously and it does not support unsymmetrical similarity. Therefore, in our experiments we have been forced to use only classic indiscernibility and symmetrical similarity as components of attribute constrained indiscernibility relation and special data preprocessing.

Table 1. Number of experiments for each data set using ten-fold cross-validation (CV10) and all possible to use attribute constrained indiscernibility relations (2^n, where n is number of attributes).

Dataset name	ech	hin	hur2	pid2	smo2
Number of experiments	640	640	640	2560	2560

The data used in experiments was preprocessed, in order to enable simultaneous utilization of classic indiscernibility relation and symmetrical similarity. The algorithm for decision rule induction was parameterized to use the symmetrical similarity and all missing attribute values in columns (attributes), that should be processed utilizing symmetrical similarity, were left as "missing". Other missing values, that should be processed utilizing classic indiscernibility were replaced by special unused domain value ("-9999"). Thanks to such a data preprocessing we are able to simulate simultaneous use of both classical indiscernibility and symmetrical similarity in the same decision table.

5.3. Empirical Evaluation

The aim of the experimental evaluation was to compare how well heuristical function $h'(R)$ selects an indiscernibility relation that gives the best classification accuracy. We carried out two groups of experiments. In the first group of experiments the $h'(R)$ value was computed for each training data and for each possible attribute constrained indiscernibility relation. In the second group of experiments the decision rules were induced from training data for each possible attribute constrained indiscernibility relation and the actual classification accuracy was calculated over the test data. The number of experiments in each group is presented in Table 1.

The general observation from these experiments is that calculation of the heuristical function $h'(R)$ for each possible attribute constrained indiscernibility relation take less time than decision rule induction from the same data. Obviously, we tested only data tables with reasonable small number of attributes, so for wider data tables the exponential growth of possibilities will make such exhaustive calculations unfeasible. Although, the heuristical function $h'(R)$ proved to be reasonably fast for applying to such problems.

The results of classification accuracy are presented in Table 2. In the first column the average classification accuracy is presented over all ten folds of cross-validation and all attribute constrained indiscernibility relations. In the second column the classification accuracy is averaged only over the indiscernibility relations that have maximal value of heuristic $h'(R)$ for considered fold of cross-validation. To better compare these results we provide also further comparison. When we know all classification results, then a posteriori we can select which indiscernibility relation is indeed the best. We averaged classification results from all ten folds and selected only relations that give the highest average accuracy. In Table 2, in the third column the results are averaged only over relations that are the best in average. The main difference between relations selected by heuristic function and relations the best in average consist in that the relations selected by heuristic are selected individually for each fold of cross-validation, while relations the best in average are selected for all folds.

As we can see in Table 2, the indiscernibility relations selected by heuristic $h'(R)$ achieves higher average accuracy than all indiscernibility relations in four of five cases. Another interesting observation

Table 2. Averaged results of classification accuracy over all ten folds of cross-validation and all or selected indiscernibility relations.

Dataset name	Average accuracy	Average accuracy over selected relations	Average accuracy over the best relations
ech	64.35 ±13.39	65.69 ±12.49	70.50 ±14.35
hin	61.22 ±4.80	64.00 ±3.36	63.70 ±3.93
hur2	79.78 ±7.05	78.47 ±7.71	81.79 ±6.91
pid2	71.43 ±4.71	74.10 ±4.52	72.79 ±3.06
smo2	53.00 ±2.17	54.33 ±2.17	54.33 ±2.17

is that indiscernibility relations the best in average not always achieves the highest average accuracy. The first observation suggest that indeed heuristic $h'(R)$ usually selects good indiscernibility relations, although not always. The second observation can be explained by the fact, that indiscernibility relation the best for all ten folds of cross-validation need not be necessary the best for a particular fold. The relations selected by heuristic $h'(R)$ are individually chosen for each fold of cross-validation, so it is possible that they achieve better classification accuracy on separate folds of cross-validation. However, it raise doubts whether this fact should be perceived as improvement or rather as over-fitting.

The numbers of selected relations are presented in Table 3. In the first column the total number of all possible attribute constrained indiscernibility relations is presented in all ten folds of cross-validation. For datasets with six attributes it is $2^6 \cdot 10 = 640$, while for the datasets with eight attributes it is $2^8 \cdot 10 = 2560$. In the second column the number of relations the best for all ten folds is presented. Such relation need not be uniquely identified. In our case there are two, sixteen or even sixty four equally good relations. In the third column is presented the number of relations selected by heuristic $h'(R)$. Again in this case heuristic $h'(R)$ can select more than one relation for each fold, but this time also this number need not be divisible by ten, while the number of selected relations can differ from fold to fold. We observe such situation for dataset ech. We can observe also that the number of relations selected by heuristic $h'(R)$ is very similar to the number of relations the best for all ten folds. Nevertheless, these relations are not the same ones, what suggest the last column. In the fourth column is presented the number of relations that are simultaneously the best for all ten folds and selected by heuristic $h'(R)$. As we can see, for all datasets except the last one the intersection of these two groups of relations is empty. We do not expect that the similar number of relations in the second and the third column is incidental. It should have something in common with the fact, that some differently defined indiscernibility relations contain exactly the same pairs of objects. Although, empty intersection of these two groups of relations for most of the datasets suggest, that the underlying rule for that is not so trivial.

6. Conclusions

The research on flexible indiscernibility relations has been inspired by weaknesses of standard indiscernibility relations observed by Stefanowski (see, e.g., [31]), experience in developing the decomposi-

Table 3. Number of all and selected indiscernibility relations summed in all ten folds of cross-validation.

Dataset name	Total number of relations	Number of the best relations	Number of selected relations	Number of the best and selected relations
ech	640	20	22	0
hin	640	20	20	0
hur2	640	160	160	0
pid2	2560	160	160	0
smo2	2560	640	640	640

tion method (see, e.g., [14]) and modelling of indiscernibility relation minimizing inconsistencies using CAKE methodology (cf. [15]). This paper is an extended version of [16]. Independently a similar approach has been developed by Grzymała-Busse (see, e.g., [8, 9]), where very similar approach to the flexible indiscernibility relations is proposed. The main difference of that work is the assumption that missing attribute values are notated using two different symbols, which enables applying two semantics of missing attribute values. Under such an approach semantics of missing value is a priori given and searching for optimal indiscernibility relation is not considered. The important contribution of that work is a complete algorithm for decision rule induction using different semantics of missing attribute values. Although, proposed there MLEM2 algorithm is not able to deal with general flexible indiscernibility relations presented here.

The presented two approaches for constructing flexible and customizable for a considered information system indiscernibility relations provide a foundation for considering the problem of fitting an indiscernibility relation to an information system. The flexibility in selecting any indiscernibility relation between classic indiscernibility relation and symmetrical similarity is limited by some assumptions. This property provides an opportunity to efficiently search for optimal solution for this problem. The goal of introducing these relations is improvement in reasoning from data with missing attribute values.

The attribute constrained indiscernibility relation is simpler in its construction and is limited by much stronger assumption. From one point of view this gives less flexibility, but from another it simplifies construction of an algorithm that search for such a relation. The descriptor constrained indiscernibility relation is more complex as it is limited by weaker assumption. This gives a lot of flexibility, but makes the searching for an (sub)optimal relation more difficult. Perhaps the efficient algorithms that search for descriptor constrained indiscernibility relation will be searching only in a special family of such relations, to keep the computations in reasonable time.

To get a complete solution for this problem we have to provide an algorithm that searches for an optimal indiscernibility relation. This paper tries to initiate this work for attribute constrained indiscernibility relations, but there is still a lot of work to do in the case of descriptor constrained indiscernibility relations. Two possible approaches for further work can be learning relations from data tables (cf. [21]) or applying imputation algorithms as schemes for transcription into flexible indiscernibility relations (cf. [6]).

The other issue is related with classifier induction. Most of the rough set concepts, such as lower and upper approximations or reducts, are naturally extensible to the case with an arbitrary indiscerni-

bility relation. However, even if the classifier induction algorithms, in particular decision rule induction algorithms (see, e.g., [11, 30]) are easily extensible to the case with an arbitrary indiscernibility relation, their current implementations are fixed to one or two specific indiscernibility relations (see, e.g., [2, 8, 9]. Apart from that, there are several decision rule induction algorithms that implicitly utilizes classic indiscernibility relation or symmetrical similarity. Therefore it is necessary to extend decision rule induction algorithms to the case with an arbitrary indiscernibility relation.

Acknowledgments

The author would like to thank professor Andrzej Skowron for his support in this research. The research has been supported by the grant 3T11C00226 from Ministry of Scientific Research and Information Technology of the Republic of Poland.

References

[1] Alpigini, J. J., Peters, J. F., Skowron, A., Zhong, N., Eds.: *Rough Sets and Current Trends in Computing, Third International Conference, RSCTC 2002*, LNCS 2475, Springer, 2002.

[2] Bazan, J. G., Szczuka, M., Wojna, A., Wojnarski, M.: On the Evolution of Rough Set Exploration System., *Rough Sets and Current Trends in Computing, RSCTC 2004* (S. Tsumoto, R. Słowiński, H. J. Komorowski, J. W. Grzymała-Busse, Eds.), LNCS 3066, Springer, 2004.

[3] Candan, K. S., Grant, J., Subrahmanian, V. S.: A Unified Treatment of Null Values using Constraints, *Information Sciences*, **98**(1-4), 1997, 99–156.

[4] Codd, E. F.: Understanding Relations (Installment #7), *FDT - Bulletin of ACM SIGMOD*, **7**(3/4), 1975, 23–28.

[5] Fujikawa, Y., Ho, T. B.: Scalable Algorithms for Dealing with Missing Values, 2001.

[6] Gediga, G., Düntsch, I.: Maximum Consistency of Incomplete Data via Non-Invasive Imputation, *Artificial Intelligence Review*, **19**, 2003, 93–107.

[7] Greco, S., Matarazzo, B., Słowiński, R.: Fuzzy Similarity Relation as a Basis for Rough Approximations, *Rough Sets and Current Trends in Computing, RSCTC'98* (L. Polkowski, A. Skowron, Eds.), LNCS 1424, Springer, 1998.

[8] Grzymała-Busse, J. W.: Rough set strategies to data with missing attribute values, *Proc. of the Workshop on Foundations and New Directions in Data Mining, associated with ICDM-2003*, 2003.

[9] Grzymała-Busse, J. W.: Data with missing attribute values: Generalization of idiscernibility relation and rule induction, *Transactions on Rough Sets 1*, LNCS 3100, Springer, 2004.

[10] Grzymała-Busse, J. W., Hu, M.: A Comparison of Several Approaches to Missing Attribute Values in Data Mining, in: Ziarko and Yao [34], 378–385.

[11] Komorowski, J., Pawlak, Z., Polkowski, L., Skowron, A.: Rough Sets: A Tutorial, *Rough Fuzzy Hybridization. A New Trend in Decision Making* (S. K. Pal, A. Skowron, Eds.), Springer, Singapore, 1999.

[12] Komorowski, J., Polkowski, L., Skowron, A.: Learning Tolerance Relations by Boolean Descriptors: Automatic Feature Extraction from Data Tables, *RSFD'96* (S. Tsumoto, et al., Eds.), 1996.

[13] Kryszkiewicz, M.: Properties of Incomplete Information Systems in the Framework of Rough Sets, in: Polkowski and Skowron [23], 422–450.

[14] Latkowski, R.: Incomplete Data Decomposition for Classification, in: Alpigini et al. [1], 413–420.

[15] Latkowski, R.: Optimal indiscernibility relation for missing attribute values using CAKE (in Polish), 2002.

[16] Latkowski, R.: On Indiscernibility Relations for Missing Attribute Values, *CS&P'2004, Volume 2. Informatik-Bericht Nr. 170* (e. a. G. Lindemann, Ed.), Humboldt University, 2004.

[17] Lim, T.: *Missing covariate values and classification trees*, http://www.recursive-partitioning.com/mv.shtml, Recursive-Partitioning.com, 2000.

[18] Lipski, W. J.: On Semantic Issues Connected with Incomplete Information Databases, *ACM Transactions on Database Systems*, **4**(3), 1979, 262–296.

[19] Little, R. J. A., Rubin, D. B.: *Statistical Analysis with Missing Data*, John Wiley and Sons, 1987.

[20] Nguyen, H. S.: *Discretization of real value attributes. Boolean reasoning approach*, Ph.D. Thesis, Warsaw University, Faculty of Mathematics, Computer Science and Mechanics, 1997.

[21] Nguyen, S. H.: *Regularity Analysis and its Application in Data Mining*, Ph.D. Thesis, Warsaw University, Faculty of Mathematics, Computer Science and Mechanics, 1999.

[22] Pawlak, Z.: *Rough sets: Theoretical aspects of reasoning about data*, Kluwer, Dordrecht, 1991.

[23] Polkowski, L., Skowron, A., Eds.: *Rough Sets in Knowledge Discovery 1: Methodology and Applications*, Physica-Verlag, 1998.

[24] Polkowski, L., Skowron, A., Żytkow, J. M.: Tolerance Based Rough Sets, *Soft Computing* (T. Y. Lin, A. M. Wildberger, Eds.), San Diego Simulation Councils Inc., 1995.

[25] Pomykała, J. A.: About Tolerance and Similarity Relations in Information Systems, in: Alpigini et al. [1], 175–182.

[26] Skowron, A., Nguyen, H. S.: Boolean Reasoning Scheme with Some Applications in Data Mining, in: Żytkow and Rauch [35], 107–115.

[27] Ślęzak, D., Wróblewski, J.: Classification Algorithms Based on Linear Combinations of Features, in: Żytkow and Rauch [35], 548–553.

[28] Ślęzak, D., Wróblewski, J.: Application of Normalized Decision Measures to the New Case Classification, in: Ziarko and Yao [34], 553–560.

[29] Słowiński, R., Stefanowski, J.: Rough classification in incomplete information systems, *Math. Computing Modelling*, **12**, 1989, 1347–1357.

[30] Stefanowski, J.: On rough set based approaches to induction of decision rules, in: Polkowski and Skowron [23], 500–529.

[31] Stefanowski, J., Tsoukiàs, A.: On the Extension of Rough Sets under Incomplete Information, *New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, RSFDGrC '99* (N. Zhong, A. Skowron, S. Ohsuga, Eds.), LNCS 1711, Springer, 1999.

[32] Stefanowski, J., Tsoukiàs, A.: Incomplete Information Tables and Rough Classification, *International Journal of Computational Intelligence*, **17**(3), August 2001, 545–566.

[33] Wróblewski, J.: *Adaptative Methods for Object Classification (in Polish)*, Ph.D. Thesis, Warsaw University, Faculty of Mathematics, Computer Science and Mechanics, 2001.

[34] Ziarko, W., Yao, Y. Y., Eds.: *Rough Sets and Current Trends in Computing, RSCTC 2000*, LNCS 2005, Springer, 2001.

[35] Żytkow, J. M., Rauch, J., Eds.: *Principles of Data Mining and Knowledge Discovery, PKDD '99*, LNCS 1704, Springer, 1999.