Building a Rule-Based Classifier—A Fuzzy-Rough Set Approach

Suyun Zhao, *Student Member*, *IEEE*, Eric C.C. Tsang, Degang Chen, and XiZhao Wang, *Senior Member*, *IEEE*

Abstract—The fuzzy-rough set (FRS) methodology, as a useful tool to handle discernibility and fuzziness, has been widely studied. Some researchers studied on the rough approximation of fuzzy sets, while some others focused on studying one application of FRS: attribute reduction (i.e., feature selection). However, constructing classifier by using FRS, as another application of FRS, has been less studied. In this paper, we build a rule-based classifier by using one generalized FRS model after proposing a new concept named as "consistence degree" which is used as the critical value to keep the discernibility information invariant in the processing of rule induction. First, we generalized the existing FRS to a robust model with respect to misclassification and perturbation by incorporating one controlled threshold into knowledge representation of FRS. Second, we propose a concept named as "consistence degree" and by the strict mathematical reasoning, we show that this concept is reasonable as a critical value to reduce redundant attribute values in database. By employing this concept, we then design a discernibility vector to develop the algorithms of rule induction. The induced rule set can function as a classifier. Finally, the experimental results show that the proposed rule-based classifier is feasible and effective on noisy data.

Index Terms—Knowledge-based systems, fuzzy-rough hybrids, rule-based classifier, IF-THEN rule.

1 Introduction

Real-world applications in the areas of artificial intelligence, such as pattern recognition and machine learning, require that their tools be able not only to reduce dimensionality of large databases, but also build classifiers on classification problems. The fuzzy-rough set (FRS) methodology [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17] is one widely studied class of tool for reducing database dimensionality (also known as attribute reduction) and building classifiers [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33]. There have been some approaches proposed that used FRS to build classifiers but most of the resultant classifiers either required the support of other classifiers or had not been shown to have a mathematical foundation or were not robust to misclassification or perturbation [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45].

The approaches of classifier building with either Rough Sets (RSs) or FRS can be roughly divided into three categories. The first category has been to use FRS (or RS) as a preprocessing tool [35], [36], [37], [38]. For example, in [35], FRS was not used to induce rules, but merely to reduce features. This was a two-step process in which each of the steps was independent. Its first step used RS to select features and the second step built a rule-based classifier by extracting some fuzzy rules from a perceptron network classifier in the reduced space.

The second category uses RS or FRS as part of classifier-designing tool [39], [40], [41]. The fuzzy-rough classification tree classifier [41], for example, used fuzzy-rough hybrids to qualify the dependency of decision attributes in the decision tree generation mechanism.

The third category uses fuzzy-rough hybrids to construct classifiers. It does so without the assistance of other classifiers [42], [43], [44], [45]. However, the resultant classifiers in this category have some observable limitations. For example, the classifier designed using RS in [42], [43] could only operate effectively on the data sets containing discrete values and they needed perform a discretization step beforehand on the data sets containing real-valued features. Although Wang et al. [45] used RS to design a rule-based classifier without discretization and performed well on some data sets, they did not consider the theoretical structures of the lower and upper approximations, such as topologic and algebraic properties, or use them in knowledge discovery. That is, there exist one gap between the existing FRS methodology and its application of classifier building.

In this paper, the focus is on narrowing this gap by building a classifier using the FRS methodology. However, we find that it is not an effective selection of building classifier using the existing FRS since it was not a robust model in the real applications. Because we have found that the existing FRS was sensitive to misclassification and perturbation [44]. Therefore, it is necessary to design a robust classifier based on the FRS methodology.

In this paper, we propose a rule-based classifier by using a generalized FRS framework. First, we generalize FRS to a robust model named as Generalized Fuzzy-rough Sets

- *S. Zhao is with the School of Mathematics and Computer Sciences, Hebei University, Baoding, China. E-mail: zsyrose2007@yahoo.com.cn.*
- *E.C.C. Tsang is with the Faculty of Information Technology, The Macau University of Science and Technology, Taipa, Macau. E-mail: cctsang@must.edu.mo.*
- *D. Chen is with the Department of Mathematics and Physics, North China Electric Power University, China. E-mail: chengdegang@263.net.*
- *X. Wang is with the School of Mathematics and Computer Sciences, Hebei University, Baoding, China. E-mail: xizhaowang@ieee.org.*

Manuscript received 14 Mar. 2008; revised 17 Dec. 2008; accepted 11 Apr. 2009; published online 29 Apr. 2009.
Recommended for acceptance by B.C. Ooi.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2008-03-0145.
Digital Object Identifier no. 10.1109/TKDE.2009.118.

(denoted by GFRS), which is used as the foundation to build a rule-based classifier. Second, we propose the consistence degree as a critical point to reduce attribute values and the strict mathematical reasoning shows that this concept is reasonable by keeping it invariant to reduce attribute values. Further, we propose a discernibility vector to find all attribute value reductions and investigate the structure of attribute value reductions. The above proposed concepts, i.e., consistence degree and discernibility vector, are the main contribution of this paper. Forth, we develop some heuristic algorithms, which achieve the near-optimal attribute value reductions and near-minimal rule set, to build a rule-based classifier. Finally, we compare our proposed method with other rule-based classifiers: RS, FRS, and fuzzy decision tree. The experimental results show the feasibility and effectiveness of our proposed method on noisy data.

The rest of this paper is organized as follows: Section 2 reviews FRS. Section 3 generalizes fuzzy approximation operators to a robust model named as GFRS. Section 4 then proposes the concepts of attribute value reduction based on GFRS and designs a discernibility vector to investigate the structure of attribute value reductions. Also, it proposes a rule induction algorithm. In Section 5, the experimental comparisons between our proposed method and other rule-based classifiers are given. Section 6 concludes this paper.

2 Review of Fuzzy-Rough Sets

Theories of RSs and fuzzy sets (FSs) are distinct and complementary to handle vagueness and uncertainty [4], [16]. RS described the idea of indiscernibility between objects in a set, while FS modeled the ill-definition of the boundary of a subclass of this set. That is to say, they were not rival theories but captured two distinct aspects of imperfection in knowledge [4]. Many researchers were then inspired to combine them to handle indiscernibility and fuzziness [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. The results of these studies led to the instruction of the notions of FRS.

Generally speaking, FRS was composed of two parts. One was knowledge representation (i.e., the rough approximation of fuzzy sets), while the other was knowledge discovery (i.e., attribute reduction and rule induction). Most researchers focused on knowledge representation of FRS by using two approaches: the constructive and axiomatic [3], [4], [5], [6], [7], [8], [9], [10], [11], [14]. The axiomatic approach took the fuzzy approximation operators as the primary notion and focused on studying the mathematical structure of FRS, such as algebraic and topologic structures. On the contrary, the constructive approach took the replacements of the equivalence relation, such as fuzzy binary relation, fuzzy T-similarity relation, and fuzzy weak equivalence partition, as the primary notion. Other notions, such as the lower and upper approximations, were constructed by using the replacements.

The summarization of FRS has been done in [1], [9], [13], [23] and interested readers can refer to them. In this paper, we just review some researches of FRS summarized in [9], [23], which are helpful for us to build a rule-based classifier. As preliminaries, a fuzzy decision table used as the study platform of FRS is formulated as follows:

2.1 Fuzzy Decision Table

Let U be a nonempty set with finite objects. With every object, we associate a set of condition attributes R and a set of decision attributes D. The pair $(U, R \cup D)$ is called a decision table, denoted by DS. If some attributes in DS are fuzzy attributes, the decision table is called fuzzy decision table, denoted by $FD = (U, R \cup D)$. Here, fuzzy attributes mean the attributes with real number values since these values can be transferred to fuzzy values. For simplicity, we still use R to represent the set of fuzzy attributes.

In a fuzzy decision table $FD = (U, R \cup D)$, $a(x)$ represents the value of $x \in U$ on the attribute a and $P(x) = \{a(x) : a \in P \subseteq R\}$ represents the subset of attribute values of $x \in U$ on attribute subset $P \subseteq R$. With every $P \subseteq R$, we associate a binary relation $P(x, y)$ called fuzzy similarity relation of P, which is a binary relation satisfying reflexivity ($P(x, x) = 1$), symmetry ($P(x, y) = P(y, x)$), and $T-$ transitivity ($P(x, y) \geq T(P(x, z), P(z, y))$) for every $x, y, z \in U$. Actually, when the attribute values are symbolic, the fuzzy similarity relation degenerates to an equivalence relation, which generates a partition on U, denoted by $U/P = \{[x]_P \mid x \in U\}$, where $[x]_P = \{y \in U \mid a(x) = a(y), \forall a \in P\}$ is the equivalence class containing $x \in U$.

In most practical applications, condition attributes often have real number values, while decision attributes have symbolic values. This type of fuzzy decision table with condition fuzzy attributes and decision symbolic attributes is thus acted as the platform to design the rule-based classifier by using FRS in this paper.

2.2 Fuzzy-Rough Sets

The concept of FRS was first proposed by Dubois and Prade [3], [4], their idea was as follows: Let U be a nonempty universe and R a fuzzy binary relation on U, $F(U)$ the fuzzy power set of U. A fuzzy-rough set is a pair $(R_*(F), R^*(F))$ of a fuzzy set F on U such that for every $x \in U$,

$$R_*(F)(x) = inf_{y \in U}\ max\{1 - R(x, y), F(y)\},$$
$$R^*(F)(x) = sup_{y \in U}\ min\{R(x, y), F(y)\}.$$

In [10], the above Dubois and Prade fuzzy-rough set was generalized from Max, Min to a residuated implicator ϑ and a general triangular norm T with respect to a fuzzy similarity relation R. The lower and upper approximation operators of a fuzzy set A are defined as for every $A \in F(U)$,

$$\underline{R}A(x) = inf_{u \in U}\ \vartheta(R(u, x), A(u)),$$
$$\overline{R}A(x) = sup_{u \in U}\ T(R(u, x), A(u)).$$

However, another upper approximation operator was proposed to obtain the dual approximation operator of $\underline{R}A$ since $\overline{R}A$ and $\underline{R}A$ were not dual with each other [5]. Suppose S is the dual triangular conorm of T, define $\overline{R}(A)(x) = sup_{y \in U}\ \sigma(1 - R(x, y), A(y))$ as another kind of upper approximation of a fuzzy set A.

The aforementioned approximation operators can be summarized as the following four general fuzzy approximation operators. For every $A \in F(U)$,

- T—upper approximation operator:

$$\overline{R_T}A(x) = sup_{u \in U}\ T(R(x, u), A(u));$$

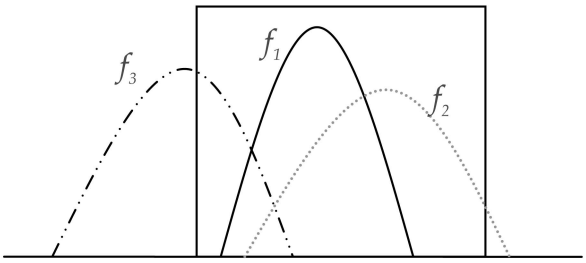

Fig. 1. The demonstration graph of the sensitivity of the existing FRS.

- S—lower approximation operator:

$$\underline{R_S}A(x) = inf_{u\in U}\ S(N(R(x,u)), A(u));$$

- σ—upper approximation operator:

$$\overline{R_\sigma}A(x) = sup_{u\in U}\ \sigma(N(R(x,u)), A(u));$$

- ϑ—lower approximation operator:

$$\underline{R_\vartheta}A(x) = inf_{u\in U}\ \vartheta(R(x,u), A(u)).$$

This type of definition is seen as the membership function representation of fuzzy approximation operators since they are defined by using membership function.

From the viewpoint of granular computing, $\overline{R_T}A$ and $\underline{R_\vartheta}A$, and $\overline{R_\sigma}A$ and $\underline{R_S}A$ are two pairs of approximation operators, respectively, since they can be represented by their individual fuzzy granules: $\overline{R_T}x_\lambda$ and $\overline{R_\sigma}x_\lambda$, where a fuzzy point x_λ is defined as [8]

$$x_\lambda(z) = \begin{cases} \lambda, & z = x, \\ 0, & z \neq x, \end{cases} \quad \forall z \in U.$$

Their granular representation is presented as follows [8]:

$$\underline{R_\vartheta}A = \cup\{\overline{R_T}x_\lambda : \overline{R_T}x_\lambda \subseteq A\}; \quad \overline{R_T}A = \cup\{\overline{R_T}x_\lambda : x_\lambda \subseteq A\};$$
$$\underline{R_S}A = \cup\{\overline{R_\sigma}x_\lambda : \overline{R_\sigma}x_\lambda \subseteq A\}; \quad \overline{R_\sigma}A = \cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A\}.$$

It is easy to see that there are two conditions $\overline{R_T}x_\lambda \subseteq A$ and $\overline{R_\sigma}x_\lambda \subseteq A$ in the granular representation of fuzzy approximation operators. These conditions control which fuzzy sets $\overline{R_T}x_\lambda$ and $\overline{R_\sigma}x_\lambda$ can be included in the fuzzy lower approximation operators $\underline{R_\vartheta}A$ and $\underline{R_S}A$. Small perturbation of $\overline{R_T}x_\lambda$ and $\overline{R_\sigma}x_\lambda$ will change the result of $\underline{R_\vartheta}A$ and $\underline{R_S}A$. These show that the conditions $\overline{R_T}x_\lambda \subseteq A$ and $\overline{R_\sigma}x_\lambda \subseteq A$ are too harsh to construct the fuzzy lower approximation operators $\underline{R_\vartheta}A$ and $\underline{R_S}A$, and then, the existing FRS is sensitive to misclassification and perturbation.

In Fig. 1, we use one demonstration graph to show the sensitivity of the existing FRS. In $\overline{R_\sigma}x_\lambda \subseteq A$ or $\overline{R_T}x_\lambda \subseteq A$, $\overline{R_\sigma}x_\lambda$ or $\overline{R_T}x_\lambda$ can be represented by a fuzzy set (e.g., f_1, f_2, and f_3 in Fig. 1) and A can be represented by a rectangle.

In Fig. 1, we find that there are three kinds of inclusion relations: f_1 is included in A, most part of f_2 is included in A, and a small part of f_3 is included in A. In fact, f_2 should be included in A if we ignore some small membership degrees of f_2 (these small membership degrees may be caused by misclassification and perturbation). All these show that two conditions $\overline{R_T}x_\lambda \subseteq A$ and $\overline{R_\sigma}x_\lambda \subseteq A$ are sensitive to misclassification and perturbation. As a result, the knowledge representation power of FRS is weak.

3 Generalization of Fuzzy-Rough Sets (GFRSs)

To make FRS less sensitive to misclassification and perturbation, we generalize the existing fuzzy approximation operators as membership function representation and granular representation. We then prove that these two representations are equivalent. The membership representation of the fuzzy approximation operators in FRS is generalized by introducing a threshold, described as follows:

Definition 3.1 (membership function representation). *For every $A \in F(U)$ and a given threshold $\alpha \in [0,1)$, fuzzy lower and upper approximations are, respectively, defined as follows:*

$$\underline{R_\vartheta}_\alpha A(x) = \inf_{A(u)\le\alpha} \vartheta(R(x,u),\alpha) \wedge \inf_{A(u)>\alpha} \vartheta(R(x,u), A(u)), \forall x \in U; \quad \text{(D3.1.1)}$$

$$\overline{R_T}_\alpha A(x) = \sup_{A(u)\ge N(\alpha)} T(R(x,u), N(\alpha)) \vee \sup_{A(u)<N(\alpha)} T(R(x,u), A(u)), \forall x \in U; \quad \text{(D3.1.2)}$$

$$\underline{R_S}_\alpha A(x) = \inf_{A(u)\le\alpha} S(N(R(x,u)),\alpha) \wedge \inf_{A(u)>\alpha} S(N(R(x,u)), A(u)), \forall x \in U; \quad \text{(D3.1.3)}$$

$$\overline{R_\sigma}_\alpha A(x) = \sup_{A(u)\ge N(\alpha)} \sigma(N(R(x,u)), N(\alpha)) \vee \sup_{A(u)<N(\alpha)} \sigma(N(R(x,u)), A(u)), \forall x \in U. \quad \text{(D3.1.4)}$$

In this paper, we focus on the pair of approximation operators: $\underline{R_S}_\alpha A$ and $\overline{R_\sigma}_\alpha A$. The properties and applications of another pair of approximation operators $\overline{R_T}_\alpha A$ and $\underline{R_\vartheta}_\alpha A$ have been studied in another work [46].

Here, we take (D 3.1.3) as an example to show the difference between FRS and GFRS, which is helpful for readers to understand Definition 3.1. In FRS, the S-lower approximation operator is defined as

$$\underline{R_S}A(x) = inf_{u\in U} S(N(R(x,u)), A(u)).$$

It can be equivalently represented as (D'3.1.3):

$$\underline{R_S}A(x) = \inf_{A(u)\le\alpha} S(N(R(x,u)), A(u)) \wedge \inf_{A(u)>\alpha} S(N(R(x,u)), A(u)), \quad \text{(D'3.1.3)}$$

$$\underline{R_S}_\alpha A(x) = \inf_{A(u)\le\alpha} S(N(R(x,u)),\alpha) \wedge \inf_{A(u)>\alpha} S(N(R(x,u)), A(u)). \quad \text{(D3.1.3)}$$

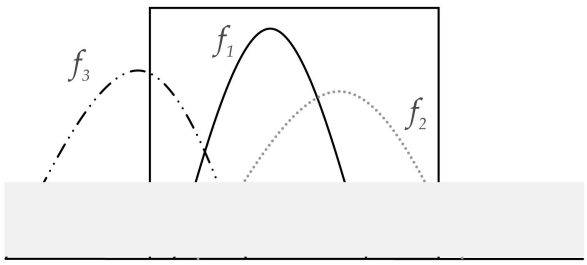

Fig. 2. The demonstration graph of the less sensitivity of GFRS.

Comparing (D 3.1.3) and (D'3.1.3), we find that in GFRS, a threshold replaces the membership degree $A(u)$ when it meets the condition $A(u) \leq \alpha$. By controlling this threshold, some misclassification and perturbation can be ignored. In the following, we would like further explain this point by using the generalized granular representation.

The generalization of the granular representation is obtained by incorporating a concept of fuzzy cut set to the granular representation of $\underline{R_S}A$ and $\overline{R_\sigma}A$. The fuzzy α-cut set is defined as

$$f^\alpha(x) = \begin{cases} f(x), & f(x) > \alpha, \\ 0, & f(x) \leq \alpha, \end{cases}$$

here, f is the fuzzy set defined on U.

Definition 3.2 (Granular representation). *For every $A \in F(U)$ and a given threshold $\alpha \in [0,1)$, fuzzy lower and upper approximations are, respectively, defined as follows:*

$$\underline{R_S}_\alpha A = \cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)^\alpha \subseteq A\}; \tag{D3.2.1}$$

$$\overline{R_\sigma}_\alpha A = \cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\}, \quad \text{where}$$
$$A_{N(\alpha)} = \begin{cases} N(\alpha), & A(x) \geq N(\alpha), \\ A(x), & otherwise. \end{cases} \tag{D3.2.2}$$

Definition 3.2 shows that a fuzzy α-cut set is introduced in the granular representation, and then, the condition $\overline{R_\sigma}x_\lambda \subseteq A$ in FRS is relaxed by $(\overline{R_\sigma}x_\lambda)^\alpha \subseteq A$ in GFRS. This condition $(\overline{R_\sigma}x_\lambda)^\alpha \subseteq A$ is less sensitive to misclassification and perturbation since small perturbation of $\overline{R_\sigma}x_\lambda$ will not change the result of $\underline{R_S}_\alpha A$. In the following, we use one demonstration graph to illustrate this point (see Fig. 2).

As compared to Fig. 1, Fig. 2 cuts those fuzzy sets f_1, f_2, and f_3 by a gray rectangle. Fig. 2 shows that after we cut some small membership degrees, f_1 and f_2 are included in A, but f_3 is not included in A. This shows that when we ignore some small misclassification and perturbation, some more fuzzy sets are chosen to compute the fuzzy lower approximation value. Thus, the fuzzy lower approximation value becomes larger, and then, the fuzzy positive region becomes larger. That is to say, the knowledge representation power becomes stronger in GFRS. All these show that GFRS is a robust framework with respect to perturbation and misclassification.

The membership function and granular representations of $\underline{R_S}_\alpha A$ and $\overline{R_\sigma}_\alpha A$ are equivalent, respectively, i.e., (D 3.1.3) and (D 3.2.1), and (D 3.1.4) and (D 3.2.2) are equivalent, respectively. We verify this fact by using the following theorems. As preliminaries, a lemma about fuzzy granular $\overline{R_\sigma}_\alpha x_\lambda$ is described as follows:

Lemma 3.1. *Let*

$$\gamma = (\cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)^\alpha \subseteq A\})(x) \tag{L3.1.1}$$

for $x \in U$, then the following statements hold: $(\overline{R_\sigma}x_\gamma)^\alpha \subseteq A$; $(\overline{R_\sigma}x_\beta)^\alpha \subseteq A$ does not hold for any $\beta > \gamma$.

$$\cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\} = \cup\{\overline{R_\sigma}x_{\lambda'} : \lambda' = A_{N(\alpha)}(x)\}. \tag{L3.1.2}$$

Proof. (L3.1.1) By $\gamma = (\cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)^\alpha \subseteq A\})(x)$, there exist $t \in (0,1]$ and $y \in U$ satisfying $\overline{R_\sigma}y_t(x) = \gamma$ and $(\overline{R_\sigma}y_t)^\alpha \subseteq A$, where $\overline{R_\sigma}y_t(x) = \sigma(N(R(y,\ x)), t)$. By $(\overline{R_\sigma}y_t)^\alpha \subseteq A$, we obtain if $\sigma(N(R(y,z)),t) > \alpha$, then $\sigma(N(R(y,z)),t) \leq A(z)$ for any $z \in U$.

$$\begin{aligned} \forall z \in U, \overline{R_\sigma}x_\gamma(z) &= \sigma(N(R(x,z)),\gamma) = \sigma(N(R(x,z)), \\ \sigma(N(R(y,x)),t)) &= \sigma(S(N(R(x,z)), N(R(y,x))),t) \\ &\leq \sigma(N(R(y,z)),t). \end{aligned}$$

Thus, we have if $\sigma(N(R(y,z)),t) > \alpha$, then

$$\sigma(N(R(x,z)),\gamma) \leq A(z),$$

i.e., $(\overline{R_\sigma}x_\gamma)^\alpha \subseteq A$ holds.

Assume that $(\overline{R_\sigma}x_\beta)^\alpha \subseteq A$ for $\beta > \gamma$, then by $\gamma = (\cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)^\alpha \subseteq A\})(x)$, we get $\gamma \geq \overline{R_\sigma}x_\beta(x) = \beta$, which contradicts the assumption $\beta > \gamma$. Thus, $(\overline{R_\sigma}x_\beta)^\alpha \subseteq A$ does not hold for $\beta > \gamma$.

(L 3.1.2). If $x_\lambda \subseteq A_{N(\alpha)}$, then $\lambda \leq A_{N(\alpha)}(x)$. Since $\sigma(\cdot,\cdot)$ is monotonically increasing in the right argument, we have $(\overline{R_\sigma}x_\lambda = \sigma(N(R(x,z)),\lambda)) \subseteq (\overline{R_\sigma}x_{\lambda'} = \sigma(N(R(x,z)), \lambda')$ for $x_\lambda \subseteq A_{N(\alpha)}$ and $\lambda' = A_{N(\alpha)}(x)$. Thus, $(\cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\}) \subseteq (\cup\{\overline{R_\sigma}x_{\lambda'} : \lambda' = A_{N(\alpha)}(x)\})$. It is straightforward to see that $(\cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\}) \supseteq (\cup\{\overline{R_\sigma}x_{\lambda'} : \lambda' = A_{N(\alpha)}(x)\})$. Hence, $(\cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\})) = (\cup\{\overline{R_\sigma}x_{\lambda'} : \lambda' = A_{N(\alpha)}(x)\})$. □

In the following, we describe and prove two theorems which show that the membership function representation and granular representation of $\underline{R_S}_\alpha A$ and $\overline{R_\sigma}_\alpha A$ are equivalent.

Theorem 3.1. *For every fuzzy set $A \in F(U)$, the following two formulas are equal:*

$$\inf_{A(u)\leq\alpha} S(N(R(x,u)),\alpha) \wedge \inf_{A(u)>\alpha} S(N(R(x,u)),A(u)), \forall x \in U; \tag{T3.1.1}$$

$$\cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)^\alpha \subseteq A\}. \tag{T3.1.2}$$

Proof. $\forall x \in U$, we need to prove

$$\inf_{A(u)\leq\alpha} S(N(R(x,u)),\alpha) \wedge \inf_{A(u)>\alpha} S(N(R(x,u)),A(u)) = \cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)^\alpha \subseteq A\}(x).$$

Let $\gamma = \cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)\alpha \subseteq A\}(x)$, then $(\overline{R_\sigma}x_\gamma)^\alpha \subseteq A$, but $(\overline{R_\sigma}x_\delta)^\alpha \subseteq A$ does not hold for $\delta > \gamma$ (by Lemma 3.1). Thus,

$$\begin{aligned}
&\forall u \in U, (\overline{R_\sigma}x_\gamma)^\alpha(u) \le A(u)\\
&\quad \Leftrightarrow \begin{cases} \sigma(N(R(x,y)),\lambda) \le A(u), & A(u) > \alpha \\ \sigma(N(R(x,y)),\lambda) \le \alpha, & A(u) \le \alpha \end{cases} \Leftrightarrow\\
&\quad \Leftrightarrow \begin{cases} inf\{c \in [0,1], S(N(R(x,y)),c) \ge \gamma\} \le A(u), & \\ \qquad\qquad\qquad\qquad\qquad\qquad A(u) > \alpha & \\ inf\{c \in [0,1], S(N(R(x,y)),c) \ge \gamma\} \le \alpha, & \\ \qquad\qquad\qquad\qquad\qquad\qquad A(u) \le \alpha \end{cases} \Leftrightarrow\\
&\quad \begin{cases} S(N(R(x,y)),A(u)) \ge \gamma, & A(u) > \alpha, \\ S(N(R(x,y)),\alpha) \ge \gamma, & A(u) \le \alpha. \end{cases}
\end{aligned}$$

Let

$$\beta = \inf_{A(u)\le\alpha} S(N(R(x,u)),\alpha) \wedge \inf_{A(u)>\alpha} S(N(R(x,u)),A(u)),$$

then we get $\beta \ge \gamma$. Assume that $\beta > \gamma$, then $(\overline{R_\sigma}x_\beta)^\alpha \subseteq A$. Thus, $\exists u'$,

$$\begin{aligned}
&(\overline{R_\sigma}x_\beta)^\alpha(u') > A(u')\\
&\quad \Leftrightarrow \begin{cases} \sigma(N(R(x,y)),\beta) > A(u'), & A(u') > \alpha \\ \sigma(N(R(x,y)),\beta) > \alpha, & A(u') \le \alpha \end{cases} \Leftrightarrow\\
&\quad \Leftrightarrow \begin{cases} S(N(R(x,y)),A(u')) < \gamma, & A(u') > \alpha, \\ S(N(R(x,y)),\alpha) < \gamma, & A(u') \le \alpha. \end{cases}
\end{aligned}$$

We get

$$\begin{aligned}
\beta = &\inf_{A(u)\le\alpha} S(N(R(x,u)),\alpha)\\
&\wedge \inf_{A(u)>\alpha} S(N(R(x,u)),A(u)) < \gamma.
\end{aligned}$$

It contradicts the assumption $\beta > \gamma$. Thus, we get $\beta = \gamma$, i.e.,

$$\begin{aligned}
&\forall x \in U, \inf_{A(u)\le\alpha} S(N(R(x,u)),\alpha) \wedge \inf_{A(u)>\alpha} S(N(R(x,u)),A(u))\\
&\quad = \cup\{\overline{R_\sigma}x_\lambda : (\overline{R_\sigma}x_\lambda)^\alpha \subseteq A\}(x).
\end{aligned}$$

□

Theorem 3.2. *For every $A \in F(U)$, the following two formulas are equal:*

$$\begin{aligned}
&\sup_{A(u)\ge N(\alpha)} \sigma(N(R(x,u)),N(\alpha))\\
&\quad \vee \sup_{A(u)<N(\alpha)} \sigma(N(R(x,u)),A(u)), \forall x \in U;
\end{aligned} \tag{T3.2.1}$$

$$\cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\}. \tag{T3.2.2}$$

Proof. By Lemma 3.1, we get $\cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\} = \cup\{\overline{R_\sigma}x_{\lambda'} : \lambda' = A_{N(\alpha)}(x)\}$. Then, we have

$$\begin{aligned}
&(\cup\{\overline{R_\sigma}u_{\lambda'} : \lambda' = A_{N(\alpha)}(u)\})(x) = sup\{\overline{R_\sigma}u_{\lambda'}(x) :\\
&\quad \lambda' = A_{N(\alpha)}(u)\}\\
&\left(\text{since } A_{N(\alpha)}(x) = \begin{cases} N(\alpha), & A(u) \ge N(\alpha) \\ A(u), & A(u) \end{cases}\right)\\
&\quad = \sup_{A(u)\ge N(\alpha)} \{\overline{R_\sigma}u_{N(\alpha)}(x)\} \vee \sup_{A(u)<N(\alpha)} \{\overline{R_\sigma}u_{A(u)}(x)\}\\
&\quad = \sup_{A(u)\ge N(\alpha)} \sigma(N(R(x,u)),N(\alpha))\\
&\quad \vee \sup_{A(u)<N(\alpha)} \sigma(N(R(x,u)),A(u)).
\end{aligned}$$

Hence, $\forall x \in U$,

$$\begin{aligned}
&\sup_{A(u)\ge N(\alpha)} \sigma(N(R(x,u)),N(\alpha))\\
&\quad \vee \sup_{A(u)<N(\alpha)} \sigma(N(R(x,u)),A(u))\\
&\quad = (\cup\{\overline{R_\sigma}x_\lambda : x_\lambda \subseteq A_{N(\alpha)}\})(x).
\end{aligned}$$

□

Theorems 3.1 and 3.2 show that the membership representation and granular representation of $\underline{R_S}_\alpha A$ and $\overline{R_\sigma}_\alpha A$ are equivalent. They are the key theorems for us to build the rule-based classifier using GFRS since they are the theoretical foundation to design the discernibility vector approach. Without them, the discernibility vector cannot be designed in GFRS.

4 Building a Rule-Based Classifier

This section aims to develop a methodology for building a rule-based classifier from a fuzzy decision table by using GFRS. This methodology is composed of two parts: first attribute value reduction, and then, rule induction from the reduced decision table. Here, the rules are IF-THEN production rules.

4.1 Attribute Value Reduction

The consistence between two objects means that the objects with similar condition attributes belong to the same decision classes. In the following, we present a theorem which describes in what condition two objects are consistent in a fuzzy decision table.

Theorem 4.1. *Given two objects x and y ($y \ne x$) in $FD = (U, R \cup D)$, if $N(R(x,y)) < \underline{R_S}[x]_D(x)$, then $[x]_D(y) = 1$.*

Proof. We prove it by contradiction. Assume that $[x]_D(y) = 0$, then $\underline{R_S}[x]_D(x) = inf_{u\in U} S(N(R(x,u)), [x]_D(u)) \le S(N(R(x,y)),0) = N(R(x,y))$. This contradicts the given condition, and then, we get $[x]_D(y) = 1$. □

Theorem 4.1 shows that if the distance between two objects is smaller than a certain value $\underline{R_S}[x]_D(x)$ ($N(R(x,y))$ can be seen as one kind of distance between the objects x and y), then these objects belong to the same decision classes. If the distance between two objects is bigger than or equal to this certain value, then they may belong to different decision classes. That is, this certain value is the critical value to keep two objects consistent. Since misclassification and perturbation are available in a fuzzy decision table, this critical value should be enlarged to adjust the robust framework of FRS.

In this paper, we enlarge the critical value to $\underline{R_{S_\alpha}}[x]_D(x)$ since it is the enlarged value of $\underline{R_S}[x]_D(x)$ in the proposed robust framework GFRS, we then obtain the following statement "if the distance between the objects x and y are smaller than the value $\underline{R_{S_\alpha}}[x]_D(x)$, then these two objects are consistent if we ignore some inconsistent objects caused by misclassification or perturbation. That is to say, if $N(R(x,y)) < \underline{R_{S_\alpha}}[x]_D(x)$, then $[x]_D(y) = 1$; if $N(R(x,y)) \geq \underline{R_{S_\alpha}}[x]_D(x)$, then $[x]_D(y) = 0$ may happen. It is clear that $\underline{R_{S_\alpha}}[x]_D(x)$ is the critical value to keep the consistence of the object x with other objects if we ignore some misclassification and perturbation. Thus, we define the consistence degree of an object in a fuzzy decision table as follows:

Definition 4.1 (consistence degree). *Given an object x in $FD = (U, R \cup D)$, let*

$$Con_{R,\alpha}(D)(x) = \underline{R_{S_\alpha}}[x]_D(x), Con_{R,\alpha}(D)(x)$$

is then called the consistence degree of x in $FD = (U, R \cup D)$.

Definition 4.1 shows that the consistence degree of each object gets its value at the lower approximation of its corresponding decision class.

Keeping the consistence degree invariant, some redundant attribute values can be reduced. That is to say, the reduction of attribute values does not cause new indiscernibility information. This allowed the fuzzy decision table to retain the main information. In the following, we use this idea in proposing the concept of attribute value reduction in $FD = (U, R \cup D)$.

Definition 4.2 (attribute value reduction). *Given an object x in $FD = (U, R \cup D)$, if the subset $P \subseteq R$ satisfies the following two statements:*

$$Con_{R,\alpha}(D)(x) = Con_{P,\alpha}(D)(x); \tag{D4.2.1}$$

$$\forall b \in P, Con_{R,\alpha}(D)(x) > Con_{\{P-\{b\}\},\alpha}(D)(x). \tag{D4.2.2}$$

Then, $P(x) = \{a(x) : a \in P\}$ is called the attribute value reduction of $x \in U$.

The attribute value $a(x) \in P(x) \subseteq R(x)$ is dispensable in $P(x)$ if $Con_{R,\alpha}(D)(x) = Con_{P-\{a\},\alpha}(D)(x)$; otherwise, it is indispensable. If all attribute values in $P(x)$ are indispensable for the object x, then $P(x)$ is independent for x.

Definition 4.3 (attribute value core). *Given an object x in $FD = (U, R \cup D)$, the collection of the indispensable attribute values in $R(x)$ is the attribute value core of x, denoted by $Core_\alpha(x)$.*

Definitions 4.2 and 4.3 show that the proposed attribute value reduction and core have similar key idea of attribute value reduction and core in RS. That is, attribute value reduction is the minimal subset keeping the discernable information of one object invariant, attribute value core is the collection of those important attribute values in an object.

In the following, we study the structure of attribute value reductions and core. As preliminaries, we present two lemmas as follows:

Lemma 4.1. *In a fuzzy decision table $FD = (U, R \cup D)$, if $(\overline{R_\sigma} x_\lambda)^\alpha \subseteq [z]_D$, then $(\overline{R_\sigma} x_\lambda)^\alpha \subseteq [x]_D$.*

Proof. If $[z]_D = [x]_D$, it is straightforward to get $(\overline{R_\sigma} x_\lambda)^\alpha \subseteq [x]_D$. If $[z]_D \neq [x]_D$, we have $\overline{R_\sigma} x_\lambda(x) \leq \alpha$ for $x \notin [z]_D$, i.e., $\lambda \leq \alpha$. When $\lambda \leq \alpha$, we get $((\overline{R_\sigma} x_\lambda)_\alpha = \phi) \subseteq [x]_D$. □

Lemma 4.2. *For two T-similarity relations R and P, if $\forall x, u \in U, R(x,u) \leq P(x,u)$, then the following statements hold: (L 4.2.1) $\underline{R_{\vartheta_\alpha}} A \supseteq \underline{P_{\vartheta_\alpha}} A$; (L 4.2.2) $\overline{R_{T_\alpha}} A \subseteq \overline{P_{T_\alpha}} A$.*

Proof.

$$\begin{aligned} \forall x, u \in U, R(x,u) \leq P(x,u) \Rightarrow & \inf_{A(u) \leq \alpha} \vartheta(R(x,u), \alpha) \\ \wedge & \inf_{A(u) > \alpha} \vartheta(R(x,u), A(u)) \geq \inf_{A(u) \leq \alpha} \vartheta(P(x,u), \alpha) \\ \wedge & \inf_{A(u) > \alpha} \vartheta(P(x,u), A(u)) \end{aligned} \tag{L4.2.1}$$

(By the antimonotonicity of $\vartheta(\cdot,\cdot)$ in the left argument) $\Rightarrow \underline{R_{\vartheta_\alpha}} A \supseteq \underline{P_{\vartheta_\alpha}} A$.

$$\begin{aligned} \forall x, u \in U, R(x,u) \leq P(x,u) \Rightarrow & \sup_{A(u) \geq 1-\alpha} T(R(x,u), 1-\alpha) \\ \wedge & \sup_{A(u) < 1-\alpha} T(R(x,u), A(u)) \\ \leq & \sup_{A(u) \geq 1-\alpha} T(P(x,u), 1-\alpha) \wedge \sup_{A(u) < 1-\alpha} T(P(x,u), A(u)) \end{aligned} \tag{L4.2.2}$$

(By the monotonicity of $T(\cdot,\cdot)$ in both arguments) $\Rightarrow \overline{R_{T_\alpha}} A \subseteq \overline{P_{T_\alpha}} A$. □

Theorem 4.2. *Given $x \in U$ and $P \subseteq R$ in $FD = (U, R \cup D)$, the following two statements are equivalent:*

- *(T 4.2.1) $P(x)$ contains the attribute value reduction of x.*
- *(T 4.2.2) The formula $\sigma(N(P(x,y)), \lambda) \leq \alpha$ always holds for every $y \notin [x]_D$, where $\lambda = \underline{R_{S_\alpha}}[x]_D(x)$.*

Proof. (T 4.2.1) $\Rightarrow$ (T 4.2.2): Assume that $P(x)$ contains the attribute value reduction of x, then $Con_{R,\alpha}(D)(x) = \underline{R_{S_\alpha}}[x]_D(x) = Con_{P,\alpha}(D)(x) = \underline{P_{S_\alpha}}[x]_D(x)$ holds. Let $\lambda = \underline{R_{S_\alpha}}[x]_D(x)$, then $\underline{P_{S_\alpha}}[x]_D(x) = \lambda$. By Lemma 4.1, $(\overline{P_\sigma} x_\lambda)^\alpha \subseteq [x]_D$ holds. Thus, $\sigma(N(P(x,y)), \lambda) \leq \alpha$ for every $y \notin [x]_D$.

(T 4.2.2) $\Rightarrow$ (T 4.2.1): By the condition $\lambda = \underline{R_{S_\alpha}}[x]_D(x)$, we get $\lambda \geq \underline{P_{S_\alpha}}[x]_D(x)$ by Lemma 4.2.

By the condition $\sigma(N(P(x,y)), \lambda) \leq \alpha$ for every $y \notin [x]_D$, we have $(\overline{P_\sigma} x_\lambda)^\alpha \subseteq [x]_D$. By the granular representation of $\underline{R_{S_\alpha}} A$, $(\overline{P_\sigma} x_\lambda)(x) = \lambda \leq \underline{P_{S_\alpha}}[x]_D(x)$ holds. Thus, $\lambda = \underline{P_{S_\alpha}}[x]_D(x) = \underline{R_{S_\alpha}}[x]_D(x)$, i.e., $Con_{R,\alpha}(D)(x) = Con_{P,\alpha}(D)(x)$. By Definition 4.2, $P(x)$ contains the attribute value reduction of the object x. □

By Theorem 4.2, we construct discernibility vector as follows:

Suppose $U = \{x_1, x_2, x_3, \ldots, x_n\}$, by $Vector(U, R, D, \alpha)$, we denote an $n \times 1$ vector (c_j), called the discernibility vector of the object x_i, such that

$$c_j = \{a : \sigma(N(a(x_i, x_j)), \lambda) \leq \alpha\}, \text{ for } D(x_i, x_j) = 0, \tag{V4.1}$$

where $\lambda = \underline{R_{S_\alpha}}[x_i]_D(x_i)$,

$$c_j = \phi, \text{ for } D(x_i, x_j) = 1. \tag{V4.2}$$

The proposed discernibility vector is similar with one column of discernibility matrix in [47]. c_j is the collection of attribute maintaining $\sigma(N(a(x_i, x_j)), \lambda) \leq \alpha$, each attribute in which can distinguish x_i from x_j.

The following theorems show that by using discernibility vector, we can find the value reductions and core.

Theorem 4.3. *$Core_\alpha(x) = \{a(x) : c_j = \{a\}\}$ for some $1 \leq j \leq n$.*

Proof.

$$a(x) \in Core_\alpha(x) \Leftrightarrow Con_{R,\alpha}(D)(x) > Con_{\{R-\{a\}\},\alpha}(D)(x)$$
$$\Leftrightarrow \underline{R_{S_\alpha}}([x]_D)(x) > \underline{(R-\{a\})_{S_\alpha}}([x]_D)(x) \Leftrightarrow .$$

Let $\lambda = \underline{R_{S_\alpha}}([x]_D)(x)$ and $t = \underline{(R-\{a\})_{S_\alpha}}([x]_D)(x)$, then these four formulas

$$\lambda > t, \overline{R_\sigma}(x)_\lambda \subseteq [x]_D, \quad \overline{(R-\{a\})_\sigma}(x)_t \subseteq [x]_D,$$

and $\overline{(R-\{a\})_\sigma}(x)_\lambda \not\subset [x]_D$ hold (By Lemma 4.1) $\Leftrightarrow$ there exists $y \in U (y \neq x)$ satisfying $\sigma(N((R-\{a\})(x,y)), \lambda) > \alpha$ and $\sigma(N((R-\{a\})(x,y)), t) \leq \alpha \Leftrightarrow \sigma(N(b(x,y)), \lambda) > \alpha$ holds for any $b \in R-\{a\}$; $\sigma(N(a(x,y)), \lambda) \leq \alpha$ holds $\Leftrightarrow c_j = \{a\}$. The statement $c_j = \{a\}$ implies that a is the unique attribute to maintain $\sigma(N(a(x,y)), \lambda) \leq \alpha$. □

Theorem 4.4. *Suppose $P \subset R$, then $P(x)$ contains one attribute value reduction of x if and only if $P \cap c_j \neq \phi$ for every $c_j \neq \phi$.*

The proof is straightforward by Theorem 4.2 and definition of c_j.

Corollary 4.1. *Suppose $P \subset R$, then $P(x)$ is a attribute value reduction of x if and only if $P(x)$ is the minimal subset of $R(x)$ satisfying $P \cap c_j \neq \phi$ for every $c_j \neq \phi$.*

Theorem 4.3 shows that the value core is the collection of single elements in the discernibility vector. Theorem 4.4 and Corollary 4.1 show that by using discernibility vector, we can find the attribute value reduction which is the minimal subset having the nonempty intersection with each nonempty entry in discernibility vector.

A discernibility function $f_{x,\alpha}(FD)$ for an object x in FD is a Boolean function of m Boolean variables $\overline{a_1}, \ldots, \overline{a_m}$ corresponding to the attributes $a_1, \ldots, a_m$, respectively, and defined as follows: $f_x(FD)(\overline{a_1}, \ldots, \overline{a_m}) = \wedge\{\vee(c_j): 1 \leq j \leq n\}$, where $\vee(c_j)$ is the disjunction of all variables $\overline{a}$ such that $a \in c_j$. Let $g_{x,\alpha}(FD)$ be the reduced disjunctive form of $f_{x,\alpha}(FD)$ obtained from $f_{x,\alpha}(FD)$ by applying the multiplication and absorption laws as many times as possible. Then, there exist l and $R_k \subseteq R$ for $k = 1, \ldots, l$ such that $g_{x,\alpha}(FD) = (\wedge R_1) \vee \cdots \vee (\wedge R_l)$, where every element in R_k only appears one time.

Theorem 4.5. *$Re\ d_\alpha(x) = \{R_1(x), \ldots, R_l(x)\}$, here, $Re\ d_\alpha(x)$ is the collection of all attribute value reductions of x, and $R_k(x)$ is the attribute value of x on attribute subset R_k for $k = 1, \ldots, l$.*

The proof is omitted since this theorem is similar to the one in [23].

Theorem 4.5 clearly shows that by using the discernibility vector, we can find all value reductions. In the following, we present the detail algorithm to find all value reductions by using the discernibility vector.

We now design an algorithm to compute all attribute value reductions for one object by using the discernibility vector approach. For $x_i \in U$ in $FD = (U, R \cup D)$, the algorithm to find all attribute value reductions of x_i is described in Algorithm 4.1. This algorithm can find all attribute value reductions of one object, but its computational complexity increases exponentially with the number of attributes (i.e., $O(|U| \times 2^{|A|})$, here, $|U|$ is the number of objects and $|A|$ is the number of attributes). In practical applications, it is not necessary to find all value reductions because finding a near-optimal one is enough. We then provide a heuristic algorithm (see Algorithm 4.2) to find one near-optimal attribute value reduction by rewriting the part covered by dot-lined box in Algorithm 4.1. The key idea of this algorithm uses the add-deletion strategy of reduction algorithm construction described in [48].

Algorithm 4.1: to find all attribute value reductions of x_i in $FD = (U, R \cup D)$

$\mathrm{Re}\,d \leftarrow \{\}$

(1) Compute discernibility vector $V = \{c_{ji}\}$ using the following three steps:

(1.1) Compute R;

(1.2) Compute $Con_R(D)(x_i)$;

(1.3) Compute c_{ji} : $c_{ji} = \{a : \sigma(N(a(x_i, x_j)), \lambda) \leq \alpha\}$ for $D(x_i, x_j) = 0$, where $\lambda = Con_R(D)(x_i)$; $c_{ji} = \phi$, for $D(x_i, x_j) = 1$;

(2) Compute $f_{x_i,\alpha}(FD) = \wedge\{\vee(c_{ji})\}$ with those $c_{ji} \neq \phi$ in $V(x_i, \alpha)$;

(3) Compute $g_{x_i,\alpha}(FD) = (\wedge \mathrm{Re}\,duct_1) \vee \ldots \vee (\wedge \mathrm{Re}\,duct_l)$ by $f_{x_i,\alpha}(FDT) = \wedge\{\vee(c_{ji})\}$;

(4) Output $\mathrm{Re}\,d = \{\mathrm{Re}\,duct_1(x_i), \ldots, \mathrm{Re}\,duct_l(x_i)\}$.

Algorithm 4.2 (Heuristic): to find one near-optimal attribute value reduction of the object x_i in $FD = (U, R \cup D)$

$\mathrm{Re}\,d \leftarrow \{\}, \mathrm{Re}\,duct \leftarrow \{\}$, $CD \leftarrow R$, $V \leftarrow \{c_{ji}\}$

(1) Compute the value core: $Core = \{a : c_{ji} = \{a\}\}$;

(2) Let $V' = V$, $V' = V' - \{v \in V' \mid v \cap Core \neq \phi \text{ or } v = \phi\}$, $\mathrm{Re}\,duct = \mathrm{Re}\,duct \cup Core$, $CD = CD - Core$;

(3) Do while $V' \neq \phi$

(3.1) Compute $frequence(a) = |\{v \in V' \mid a \in v\}|$ for each $a \in CD$;

(3.2) Select the attribute with maximum frequency b, let $\mathrm{Re}\,duct = \mathrm{Re}\,duct \cup \{b\}$, $CD = CD - \{b\}$;

(3.3) Let $V' = V' - \{v \in V' \mid v \cap \{b\} \neq \phi \text{ or } v = \phi\}$;

(4) Remove the superfluous attribute from $\mathrm{Re}\,duct$ until $\mathrm{Re}\,duct$ is the minimal subset satisfying $\mathrm{Re}\,duct \cap c_{ji} \neq \phi$ for every $c_{ji} \neq \phi$;

(5) Output $\mathrm{Re}\,d = \mathrm{Re}\,duct(x_i)$. (here $\mathrm{Re}\,duct(x_i)$ is the attribute values of x_i on attribute subset $\mathrm{Re}\,duct$).

In the following, we use one example to illustrate the method of attribute value reduction proposed in this paper.

TABLE 1
One Simple Decision Table

Attr / Obj	a	b	c	d	e	f	g	h	k	j	D
x_1	1	0.1	0.11111	0.88889	0.11111	0.11111	1	0.11111	0	0.88889	0
x_2	0.11111	0	1	0	0	1	0	1	0.77778	0.22222	1
x_3	0	0.1	1	0.11111	0.88889	0	0.66667	0.44444	0.77778	0.33333	1
x_4	0	0.1	1	0	1	0	0	1	0.88889	0.11111	1
x_5	0.88889	0.2	0	0	0.33333	0.66667	0	1	1	0	1
x_6	1	0.1	0	1	0	0	0.88889	0.11111	0.66667	0.44444	0
x_7	0	1	0	0	0	1	0.11111	0.88889	0.11111	1	1
x_8	0.11111	0.1	1	0	0.11111	1	0.11111	0.88889	0	1	0
x_9	1	0.1	0	0.33333	0.88889	0.11111	1	0	1	0	0
x_{10}	0.11111	0.9	0.22222	1	0	0.11111	1	0	0.88889	0.11111	1

TABLE 2
The Lower Approximation Value and the Consistence Degree

Obj	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
Lower approximation of decision class 0	0.88889	0	0	0	0	0.88889	0	0.77778	0.88889	0
Lower approximation of decision class 1	0	0.77778	1	1	1	0	1	0	0	0.88889
Consistence degree	0.88889	0.77778	1	1	1	0.88889	1	0.77778	0.88889	0.88889

Example 4.1. Given a simple decision table with 10 objects as follows (see Table 1), there are 10 condition fuzzy attributes $\mathrm{R} = \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, \mathrm{e}, \mathrm{f}, \mathrm{g}, \mathrm{h}, \mathrm{k}, \mathrm{j}\}$ and one decision symbolic attribute {D}. There are two decision classes: 0 and 1. The objects $\{x_1, x_6, x_8, x_9\}$ belong to class 0 and $\{x_2, x_3, x_4, x_5, x_7, x_{10}\}$ belong to class 1.

In this example, the Lukasiewicz T-conorm $S_L(x, y) = \min\{1, x + y\}$ and the negator $N(x) = 1 - x$ are selected as the $T-$conorm and negator operator to construct the $S-$lower approximation operator. The lower approximation is then specified as $\underline{R_S}_{\alpha} A(x) = inf_{A(u) \le \alpha} \min((1 - R(x, u)) + A(u)), \alpha) \wedge inf_{A(u) > \alpha} \min((1 - R(x, u)) + A(u)), 1)$. The consistence is computed by $Con_{R,\alpha}(D)(x) = \underline{R_S}_{\alpha}[x]_D(x)$. Let $\mathrm{alpha} = 0.05$, the lower approximations of each decision class and the consistence degree of each object are listed in Table 2. Table 2 shows that the consistence degree of each object gets its value at the lower approximation of its corresponding decision class.

The formula to compute the discernibility vector of x_i is $c_{ji} = \{a : \max(a(x_i, x_j) - 1 + \lambda, 0) \le \alpha\}$ for $D(x_i, x_j) = 0$, where $\lambda = \underline{R_S}_{\alpha}[x_i]_D(x_i)$; $c_{ji} = \phi$, for $D(x_i, x_j) = 1$. The discernibility vectors are listed in Table 3 (Each column in Table 3 corresponds to one discernibility vector). One element of discernibility vector is an attribute subset in which each attribute can distinguish one pair of objects. For example, the element of the first column and third row in Table 3 is an attribute subset $\{\mathrm{a}, \mathrm{c}\}$. Each attribute {a} or {c} can distinguish x_1 from x_3 without causing new inconsistence.

The attribute value core is the collection of the most important attribute values to distinguish one object from others. The attribute value core of each object is listed in Table 4. The attributes in value core is the collection of the element with single attribute in discernibility vector. For example, the value core of object x_3 is $\{(\mathrm{a}, 0)(\mathrm{f}, 0)\}$ in which the attribute subset {a,f} is the collection of the elements with single attribute in discernibility vector of x_3.

Using Algorithm 4.2, we compute one near-optimal attribute value reduction of each object (see Table 5). Table 5 shows that a fuzzy decision table has been significantly reduced after attribute value reduction.

TABLE 3
The Discernibility Vector of Every Object
(All Vectors Become a Matrix)

Obj	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
x_1	{ }	{a c d f g h k}	{a}	{a g}	{g k}	{ }	{a}	{ }	{ }	{a k}
x_2	{a c d f g h}	{ }	{ }	{ }	{ }	{c d f g h}	{ }	{k j}	{a c e f g h}	{ }
x_3	{a c}	{ }	{ }	{ }	{ }	{c d e}	{ }	{e f k}	{a c}	{ }
x_4	{a c d e g h k}	{ }	{ }	{ }	{ }	{c d e g h}	{ }	{e f k j}	{a c g h}	{ }
x_5	{d g h k j}	{ }	{ }	{ }	{ }	{d g h}	{ }	{a c k j}	{g h}	{ }
x_6	{ }	{a c d f g h}	{a c}	{a c d e}	{d}	{ }	{a d f}	{ }	{ }	{a}
x_7	{a b d f g}	{ }	{ }	{ }	{ }	{b d f}	{ }	{b c}	{a b e f g h k j}	{ }
x_8	{ }	{k j}	{f}	{f}	{c k j}	{ }	{c}	{ }	{ }	{d f g h k j}
x_9	{ }	{a c e f g h}	{a c}	{a c g h}	{g h}	{ }	{a j}	{ }	{ }	{a e}
x_{10}	{a k}	{ }	{ }	{ }	{ }	{a}	{ }	{b c d f g h k j}	{a e}	{ }

TABLE 4
The Attribute Value Core of Every Object

Obj	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
Core	{ }	{ }	{(a,0) (f,0)}	{(f,0)}	{(d,0)}	{(a,1)}	{(a,0), (c,0)}	{ }	{ }	{(a, 0.11111)}

TABLE 5
Attribute Value Reduction of Every Object

Obj	a	b	c	d	e	f	g	h	j	k	D
x_1	1	*	*	0.88889	*	*	*	*	*	*	0
x_2	0.11111	*	*	*	*	*	*	*	0.77778	*	1
x_3	0	*	1	*	*	0	*	*	*	*	1
x_4	0	*	*	*	*	0	*	*	*	*	1
x_5	*	*	0	0	*	*	0	*	*	*	1
x_6	1	*	*	1	*	*	*	*	*	*	0
x_7	0	*	0	0	*	*	*	*	*	*	1
x_8	*	*	1	*	*	*	*	*	0	*	0
x_9	1	*	*	*	*	*	1	*	*	*	0
x_{10}	0.11111	*	*	1	*	*	*	*	0.88889	*	1

* The star mark represents the attribute-value which has been reduced.

4.2 Rule Induction from the Reduced Decision Table

In a fuzzy decision table $FD = (U, R \cup D)$, each original object can be represented in the formulation of a decision rule. The corresponding rule of the object x is denoted by fd_x. The restriction of fd_x to R, denoted by $fd_x|R$, is defined by "$N(R(x, y)) < Con_{R,\alpha}(D)(x)$ on $R(x) = \{a(x) : a \in R\}$" and the restriction of fd_x to D is denoted by $fd_x|D = \wedge\{\frac{d(x)}{d}|d \in D\}$. They are called the condition and decision of fd_x, respectively. Thus, a fuzzy decision rule fd_x can be denoted by $fd_x|R \rightarrow fd_x|D$.

Example 4.2. Let us consider Example 4.1 again. We illustrate the rule representation of the object x_1 in a fuzzy decision table as follows:

Since the object x_1 belongs to "decision class 0," its lower approximation on "decision class 0" is 0.88889, and its consistence degree is 0.88889, the original decision rule corresponding to the object x_1 can be stated in the following formulation:

$fd_{x_1}|R \rightarrow fd_{x_1}|D$: if $1 - R(y, x_1) < 0.88889$ on

$$x_1 = \{(a,1),(b,0.1),\ (c,0.11111),\ (d,0.88889)(e,0.11111),\ (f,0.11111),(g,1),\ (h,0.11111),\ (k,0),\ (j,0.88889)\},$$

then the object y belongs to "decision class 0."

This rule represents that if the distance between y and x is smaller than a certain value, then y belongs to the decision class containing x. For example, since $1 - R(x_6, x_1) = 0.66667 < 0.88889$, the object x_6 belongs to "decision class 0" by using the decision rule $fd_{x_1}|R \rightarrow fd_{x_1}|D$; since $1 - R(x_2, x_1) = 1 > 0.88889$, the decision class of object x_2 cannot be predicted by using the decision rule $fd_{x_1}|R \rightarrow fd_{x_1}|D$.

This example shows that the decision rule corresponding to each original object is very trivial. It is necessary to reduce them from the viewpoint of attribute value reduction.

Definition 4.4 (reduction rule). *Given an object x and its corresponding rule $fd_x|R \rightarrow fd_x|D$ in $FD = (U, R \cup D)$, suppose $P(x) \subset R(x)$ is one attribute value reduction of x, then $fd_x|P \rightarrow fd_x|D$ is called the reduction rule of $fd_x|R \rightarrow fd_x|D$.*

This definition shows that each attribute value reduction corresponds to a reduction rule.

Example 4.3. Let us consider Example 4.1 again. Suppose one attribute value reduction of object x_1 is $P(x_1) = \{(a,1),(d,0.88889)\}$, then its corresponding reduction rule is represented as follows:

$fd_{x_1}|P \rightarrow fd_{x_1}|D$: if $1 - P(y, x_1) < 0.88889$ on $x_1 = \{(a,1),(d,0.88889)\}$ (here, $P = \{a, d\}$), then the object y belongs to "decision class 0."

This rule represents that if the object y is similar with the reduced object $x_1 = \{(a,1),(d,0.88889)\}$, then y belongs to the decision class containing x, i.e., $[x]_D$.

The covering power of each rule is measured using a concept named rule covering.

Definition 4.5 (rule covering). *Given an object x and its corresponding rule $fd_x|R \rightarrow fd_x|D$ in $FD = (U, R \cup D)$, for an arbitrary object $y \in U$, if $N(R(x,y)) < Con_{R,\alpha}(D)(x)$, then we say that $fd_x|R \rightarrow fd_x|D$ covers the object y. We also say that $fd_x|R \rightarrow fd_x|D$ covers the rule corresponding to the object y.*

If the rule $fd_x|R \rightarrow fd_x|D$ covers the object y, then the object y belongs to the decision class of x, i.e., $y \in [x]_D$, if we ignore some misclassification and perturbation.

Now, we present one theorem which shows the relation between the original decision rule and the reduction rule.

Theorem 4.6. *Given an object x and its corresponding rule $fd_x|P \rightarrow fd_x|D$ in $FD = (U, R \cup D)$, If the rule $fd_x|P \rightarrow fd_x|D$ covers one object $y \in U$, then its reduction rule $fd_x|P \rightarrow fd_x|D$ also covers the object y.*

Proof. Since $S(N(R(x,y)),0) < Con_{R,\alpha}(D)(x)$ and

$$Con_{R,\alpha}(D)(x) = Con_{P,\alpha}(D)(x),$$

we have $S(N(P(x,y)),0) \leq Con_{P,\alpha}(D)(x)$ (by $R(x,y) \leq P(x,y)$). Thus, the reduction rule $fd_x|P \rightarrow fd_x|D$ covers the object y. □

TABLE 6
One Reduced Table of Table 1

Obj	a	b	c	d	e	f	g	h	j	k	D	Consistence degree
x_2	0.11111	*	*	*	*	*	*	*	0.77778	*	1	0.77778
x_8	*	*	1	*	*	*	*	*	0	*	0	0.77778
x_1	1	*	*	0.88889	*	*	*	*	*	*	0	0.88889
x_4	0	*	*	*	*	0	*	*	*	*	1	1

* The star mark represents the attribute-value which has been reduced.

This theorem shows that the rule induction process keeps the discernibility information in the original decision table invariant. We also use this theorem to design an algorithm to find a near-minimal rule set from $FD = (U, R \cup D)$ (see Algorithm 4.3).

Algorithm 4.3: to find the near-minimal rule set

'Minimal_rule_set' ←{ };
(1) Compute the reduction rule of every original decision rule, add them into 'All_rules';
(2) Do while 'All_rules' is not empty
(2.1) Compute the'Cover_degree' of every rule in 'All_rules'.
(2.2) Add the rule 'Rule(x_i)' which has the maximum 'Cover_degree' into 'Minimal_rule_set';
(2.3) Delete the rules cover by 'Rule(x_i)' from 'All_rules';
(3) Output 'Minimal_rule_set'.

Here, "All_rules" denotes the collection of the reduction rules of every original decision rule; "Rule(x_i)" denotes one reduction rule of x_i; "Cover_degree(x_i)" denotes the number of rules which is covered by the reduction rule "Rule(x_i);" and "Minimal_rule_set" denotes the near-minimal rule set induced from the fuzzy decision table.

Example 4.4. Let us consider the table in Example 4.1 again. Using Algorithm 4.3, we obtain a near-minimal rule set as follows:

If $1 - P(y, x) < 0.77778$ on

$$x = \{(a, 0.11111),(j, 0.77778)\}$$

(here $P = \{a, j\}$), then y belongs to "decision class 1."

If $1 - P(y, x) < 0.77778$ on $x = \{(c,1),(j,0)\}$ (here, $P = \{c, j\}$), then y belongs to "decision class 0."

If $1 - P(y, x)) < 0.88889$ on $x = \{(a,1),(d,0.88889)\}$ (here, $P = \{a, d\}$), then y belongs to "decision class 0."

If $1 - P(y, x) < 1$ on $x = \{(a,0),(f,0)\}$ (here, $P = \{a, f\}$), then y belongs to "decision class 1."

This rule set can also be represented in a table (see Table 6).

The rule set obtained by using GFRS can be used as a classifier to predict the unseen objects. In the following, the rule-based classifier obtained by using GFRS is denoted by GFRSC.

5 Scalability Analysis and Experimental Comparison

In this section, we first analyze the scalability complexity of GFRSC, and then, experimentally compare it with several

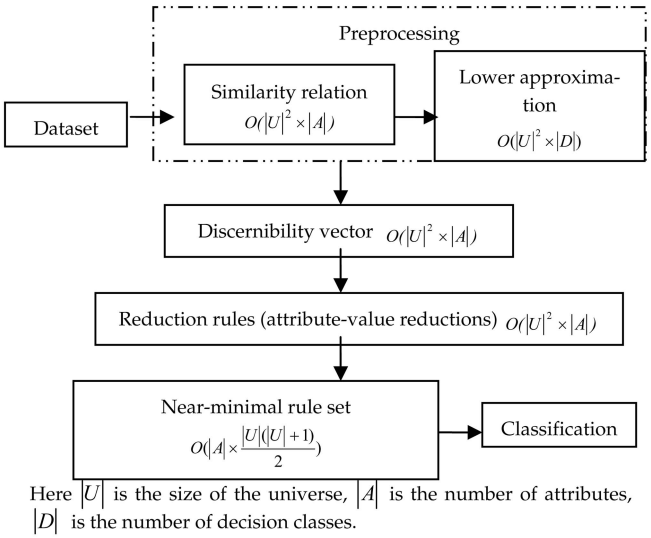

Fig. 3. The procedure and scalability analysis to rule induction.

rule-based classifiers. Since GFRS is the generalization of RS and FRS, it is necessary to compare the rule-based classifiers constructed using RS and FRS (denoted RSC and FRSC, respectively). We also compare GFRSC with one popular rule-based classifier, decision tree (denoted as DTC).

5.1 Scalability Analysis

Fig. 3 shows the procedure for inducing rules using GFRSC and an analysis of its scalability. This figure is similar to the ones for RSC and FRSC. The main difference lies in the step covered by the dot-lined box, i.e., preprocessing. In the fuzzy procedures, i.e., FRSC and GFRSC, preprocessing means computing the similarity relation and lower approximation operator. Their time and space complexities of them are $O(|U|^2 \times |A|)$ and $O(|U|^2 \times |D|)$, respectively. In the crisp case, i.e., RSC, preprocessing means the discretization of data sets. Its time and space complexity depend on the discretization method that is used.

5.2 Experimental Comparison

In this section, we experimentally compare GFRSC with some other rule-based classifiers: RSC, FRSC, and DTC. First, we specify one reasonable triangular norm for GFRSC. In GFRS (see Section 3), there are two lower approximation operators ($\underline{R_{\vartheta}}_{\alpha}$ and $\underline{R_{S}}_{\alpha}$) and two upper approximation operators ($\overline{R_{T}}_{\alpha}$ and $\overline{R}_{\sigma_\alpha}$). Generally, they are not equal to each other. If we specify the Lukasiewicz's T-norm T_L, then its dual conorm with respect to N_s is the bounded sum $S_L(x,y) = \min\{1, x+y\}$, the T_L-residuated implication is $\vartheta_L(x,y) = \min\{1, 1-x+y\}$, and the dual of ϑ_L with respect to the negator $N(x) = 1-x$ is $\sigma_L(x,y) = \max\{0, y-x\}$; clearly, we have $S_L(1-x,y) = \vartheta_L(x,y)$ and $\sigma_L(1-x,y) = T_L(x,y)$, which imply $\underline{R_{\vartheta_L}}_{\alpha} = \underline{R_{S_L}}_{\alpha}$ and $\overline{R_{(T_L)_\alpha}} = \overline{R_{(\sigma_L)_\alpha}}$ hold for a fuzzy T_L-similarity relation R and the standard negator $N(x) = 1-x$. This is one reason why we specify the Lukasiewicz's T-norm T_L in GFRSC. Furthermore, many discussions on the selection of triangular norm T emphasize the Lukasiewicz's T-norm T_L as a suitable selection [15], [53], [54]. For more on this, interested readers can refer to [24]. In the following experiments, we adopt the S-lower approximation operators constructed on T_L to design the rule-based classifier. The discernibility vector is then specified as follows:

$$c_{ji} = \{a : a(x_i, x_j) + \lambda - 1 \leq \alpha\} \quad \text{for} \quad D(x_i, x_j) = 0, \tag{V5.1}$$

where $\lambda = \underline{R_{(S_L)}}_{\alpha}([x_i]_D)(x_i)$,

$$c_{ji} = \phi, \quad \text{for} \quad D(x_i, x_j) = 1. \tag{V5.2}$$

TABLE 7
The Information of Some Data Sets from UCI Machine Learning Repository

Datasets	Abbreviation	Objects	Data type (condition attributes)	Attributes	Decision classes
Ionosphere	Ionosphere	351	Real number	34	2
Wisconsin Diagnostic Breast cancer	WDBC	569	Real number	31	2
diabetes	Diabetes	768	Real number	9	2
wine	Wine	178	Real number	14	3

The experiments are set up as follows:

Data set. Four data sets from UCI Machine Learning Repository [49] are used to compare the performance of the classifiers. The details are provided in Table 7.

Data set split. The data set is split into two parts. The randomly chosen 50 percent of objects are used as the training set to find the classifier. The remainders are used as the testing set to test the accuracy of the found classifier.

Noise control. Some noise (misclassification and perturbation) are added to the training sets since GFRSC focuses on dealing with the problems of noise (i.e., misclassification and perturbation). The percentage of the added noise $\beta \times 100\%$ means that in the randomly selected $\beta \times 100\%$ data, the decision classes are changed and condition attributes are perturbed.

Indexes. Indexes are: 1) the number of selected rules and 2) the classification accuracy of the rule set.

Parameter specification. We try the parameter in GFRSC "alpha" from 0 to 0.15 with step 0.01. We try the percentage of noise "beta" from 0 to 0.2 with step 0.05.

5.2.1 Discussion of the Parameters: Alpha and Beta

There are two parameters in the following experiments: the threshold in GFRSC "alpha" and the noise percentage "beta." In this section, we focus on analyzing the effect of alpha and beta on the classification performance of GFRSC. Figs. 4 and 5 show the classification results on four data sets. The x-axis is the value of alpha. The y-axis in Fig. 4 is the number of rules and the y-axis in Fig. 5 is the classification accuracy. There are five curves in each subfigure which correspond to the classification results on different beta values. Note that the points on y-axis is the accuracy (or the number of rules) of FRSC (Since GFRS degenerates to the classifier on FRS when alpha is set to 0), which provide some base points to evaluate the performance of GFRSC.

Fig. 4 shows that the number of rules curves with the change of alpha and beta. We find that the number of rules monotonously decreases with the value of alpha. That is, the

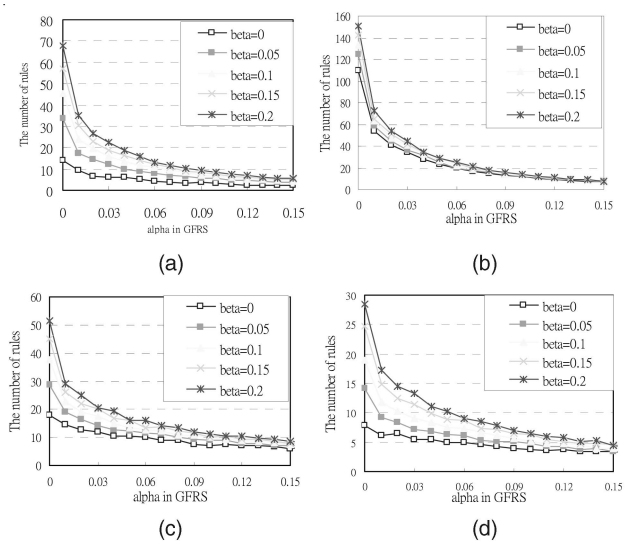

Fig. 4. The valuation of number of rules of GFRSC on different values of "alpha" and "beta." (a) WDBC. (b) Diabetes. (c) Ionosphere. (d) Wine.

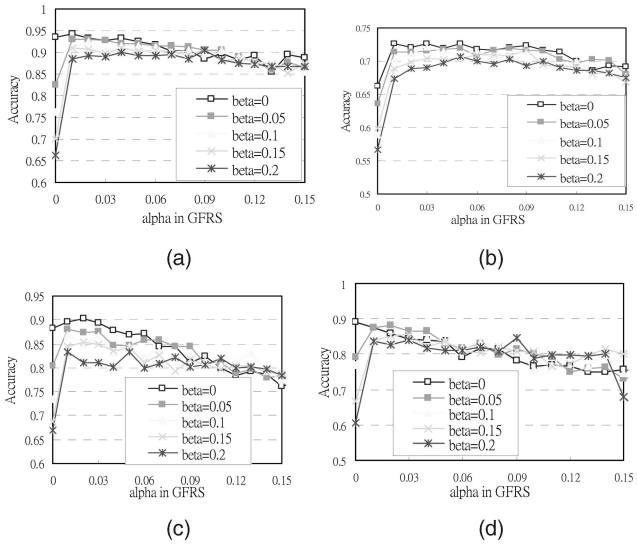

Fig. 5. The valuation of number of rules of GFRSC on different values of "alpha" and "beta." (a) WDBC. (b) Diabetes. (c) Ionosphere. (d) Wine.

bigger the parameter in GFRSC is, the compacter the obtained rule set in GFRSC is. What is more, we find that on the y-axis, the number of rules changes dramatically with the percentage of noise (i.e., beta), whereas it changes slightly when the value of alpha is large. This shows that from the viewpoint of number of rules, FRSC (i.e., $\mathrm{alpha} = 0$) is sensitive to the percentage of noise, whereas GFRSC is less sensitive.

Fig. 5 shows that the accuracy of GFRSC curves with the change of alpha and beta. We find that in most cases, the accuracy curves of GFRSC share a similar pattern. At the beginning, the accuracies increase dramatically, and then, change slightly. But when the value of alpha is too large, the accuracy decreases perceptibly. This shows that within a certain extent of alpha, the accuracy is stable and reasonable. The extent of alpha depends on the domain of specific practical problems. What is more, we also find that on y-axis, the accuracies change dramatically with the percentage of noise (i.e., beta), whereas it changes slightly when the value of alpha is large. These show that from the viewpoint of accuracy, FRSC is sensitive to the percentage of noise; whereas GFRSC is less sensitive.

The above analyses show that the noise in the data sets significantly affects the performance of FRS, whereas it has slight influence in GFRS. The basic reason is that the noise (i.e., misclassification and perturbation) is effectively controlled in the step of knowledge representation in GFRS: misclassification or small perturbation is ignored by controlling the threshold α in the fuzzy cut set $(\overline{R_\sigma} x_\lambda)^\alpha$ since the threshold α controls the accuracy of $\overline{R_\sigma} x_\lambda$ included in A.

5.2.2 Experimental Comparison GFRSC with RSC and FRSC

First of all, we would like to list the main differences among GFRS, RS, and FRS.

1. FRS and GFRS can handle real-valued data sets, whereas RS can only deal with symbolic data sets. Therefore, RS need a preprocessing of discretization.

   The selection of the discretization method affects the performance of RSC. Generally, there are two types of discretization methods for RS. One is to keep the discernibility information in the data sets invariant before and after discretization (e.g., the discretization method proposed by Nguyen which preserves the discernibility information in the original data sets [57]), while the other is a common discretization method (e.g., fuzzy C-mean). In this paper, we select fuzzy C-mean as the discretization method since some perturbation in the real-valued data sets can be omitted by it. That is to say, RSC is less sensitive to perturbation in this paper.
2. FRS is one special case of GFRS. The main difference between them is the threshold introduced into the fuzzy approximation operators. By controlling this threshold, some misclassification and perturbation are ignored in the step of knowledge representation. That is to say, GFRS is less sensitive to misclassification and perturbation.

In this section, we experimentally compare GFRSC with RSC and FRSC. Here, we focus on comparing the sensitivity, the number of rules, and classification accuracy of the obtained classifier. The comparison results are summarized in Figs. 6 and 7. The x-axis in Figs. 6 and 7 is the noise percentage, i.e., beta. The y-axis in Fig. 6 is the number of rules and the y-axis in Fig. 7 is the classification accuracy. In each subfigure, there are seven curves which correspond to the classification results on different classifiers, i.e., RSC, FRSC, and GFRSC on different values of alpha.

Figs. 6 and 7 show that in most cases, the number of rules and accuracy of RSC and FRSC increase significantly with the increment of noise, whereas the number of rules and accuracy of GFRSC increase slightly. Except the data set "Diabetes," other three data sets share these patterns. This shows that FRSC and RSC are sensitive to noise, whereas GFRSC is less sensitive.

Fig. 6 shows that GFRSC finds a compacter rule set than FRSC and RSC. It also shows that in most

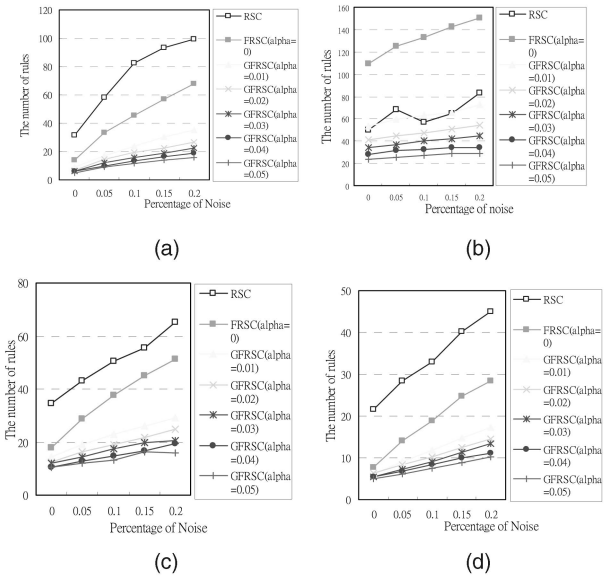

Fig. 6. The valuation of number of rules on GFRSC, RSC, and FRSC. (a) WDBC. (b) Diabetes. (c) Ionosphere. (d) Wine.

cases, the curves obtained by GFRSC are lower than the curves obtained by FRSC and RSC in most cases. That is to say, from the viewpoint of number of rules, the larger the percentage of noise is, the better the performance of GFRSC over FRSC and RSC.

Fig. 7 shows that GFRSC finds a higher accuracy than FRSC and RSC. We find that in most cases, the accuracy curves of GFRSC are often higher than those in FRSC and RSC. The larger the percentage of noise is, the greater the accuracy of GFRSC compared with FRSC and RSC.

All the above analyses show that GFRSC performs better than FRSC and RSC in terms of the noise sensitivity, number of rules, and accuracy.

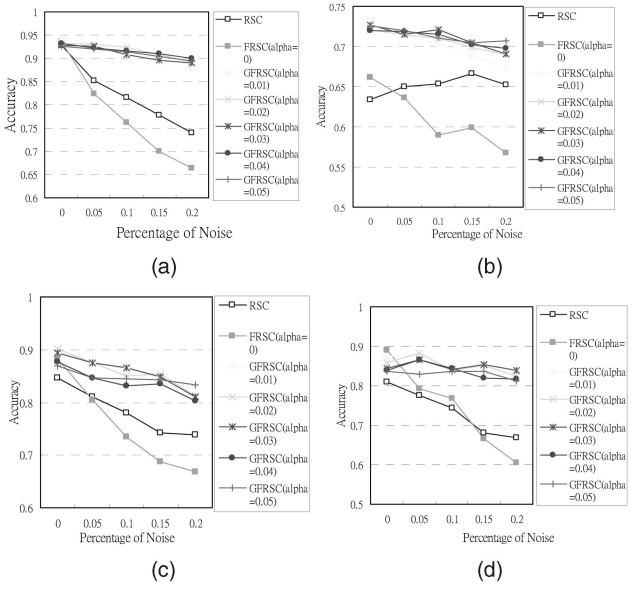

Fig. 7. The valuation of accuracy on GFRSC, RSC, and FRSC. (a) WDBC. (b) Diabetes. (c) Ionosphere. (d) Wine.

TABLE 8
The Valuation of the Comparison of DTC and GFRSC

dataset	*Beta (noise parameter)*	*DTC (see5)* Number of rules	*DTC (see5)* accuracy	*GFRSC* Number of rules	*GFRSC* accuracy
Diabetes	0	10.3	0.7362	10.095	0.71429
	0.05	9.35	0.7082	9.2857	***0.70337***
	0.10	8.55	0.6832	8.2857	***0.68539***
	0.15	7.85	0.6248	7.9524	***0.6684***
	0.2	6.4	0.62195	6	***0.67423***
Ionosphere	0	6.7	0.8922	6.619	0.78707
	0.05	7.4	0.8434	7.2381	0.78057
	0.10	6.85	0.7708	6.7619	***0.77408***
	0.15	6.7	0.73715	6.9048	***0.7757***
	0.2	8.3	0.7138	8.0952	***0.78057***
WDBC	0	4.8	0.9314	5	0.92615
	0.05	6.15	0.8737	6.381	***0.91328***
	0.10	6.5	0.81605	6.7143	***0.90459***
	0.15	4.85	0.77755	4.9524	***0.85246***
	0.2	5.3	0.72115	5.381	***0.8675***
Wine	0	4.7	0.90065	4.5714	0.81594
	0.05	5.6	0.84825	5.2857	0.82986
	0.10	5.4	0.80065	5.619	***0.80578***
	0.15	7.2	0.69715	7.0952	***0.82183***
	0.2	8.35	0.6372	8.4762	***0.82076***
Average		6.921875*	0.742188*	7.089288*	***0.791148****

* It is calculated by omitting the values on beta=0.

5.2.3 Experimental Comparison GFRSC with Fuzzy Decision Tree

Decision tree is a very popular rule-based classifier (denoted as DTC) [50], [51], [52]. It is necessary to compare our proposed rule-based classifier GFRSC with DTC. See5, as the successor of the successful and widely used ID3 and C4.5 systems (fuzzy decision tree learning algorithm) is selected to compare with GFRSC since it can deal sensibly with missing values and is pruning to deal with noisy data.

DTC and GFRS are two very different types of classifier building systems: 1) DT obtains a set of rules by constructing one tree; GFRS obtains a set of rules by reducing some redundant attribute values. 2) DTC controls the noise by pruning after the tree has been built; GFRSC controls the noise before reducing the redundant attribute values, i.e., in the first step of knowledge representation.

The classification results of GFRSC which have the similar number of rules with DTC are selected to compare with DTC. The comparison results are summarized in Table 8.

Table 8 shows that GFRSC performs better than DTC in the noised data. The bold and italic texts in Table 8 are the values of accuracy of GFRSC which are noticeably higher than DTC. For example, GFRSC significantly performs better than DTC on the noised data set "WDBC" and "Wine." The reason may be that DTC controls the noise in the last step, whereas GFRSC controls the noise in the first step.

The computational complexity of DTC is $O(|U| \times |A| \times |T|)$ (here, $|T|$ is the size of the induced tree), whereas the one of GFRSC is $O(|U|^2 \times |A|)$. The strength of DTC is that it performs well on the data sets without noise and executes faster than GFRSC. The strength of GFRSC is that it is a

robust model because it is less sensitive to misclassification and perturbation. These two classifiers suit different types of real applications.

6 Conclusion and Future Work

In this paper, we develop a rule-based classifier using a novel framework GFRS. We first propose a robust model of FRS (i.e., GFRS), and then, design a classifier building method based on GFRS. The key idea of building the classifier is to keep the discernibility information in the original data sets invariant. The proposed GFRS model, as the foundation of classifier building, has a sound mathematical rational. We, by using the strict mathematical reasoning, not only show the reasonability of the new concept "consistence degree," but also design a discernibility vector. This is the main contribution of this paper. One advantage of this paper is that the classifier GFRSC is robust since GFRS is a robust with respect to misclassification and perturbation. The main advantage of GFRS is that knowledge representation and knowledge discovery are closely related. That is, the fuzzy lower approximation operator (i.e., consistence degree) is used in attribute value reduction. This point distinguishes GFRSC from other classifiers built by using FRS.

In the future, we would like to do some more work on GFRS. One of our future work is to give a more detailed discussion on the parameter in GFRS. Now, one guideline of specifying this parameter is that in a certain reasonable extent, the larger the parameter in GFRSC is, the compacter the obtained rule set is. However, it is difficult to specify the extent of its parameter.

A discussion on GFRS and probability theory is our another future work. Our proposed GFRS model intends to solve the problem about data perturbation. If data perturbation is due to randomness, then another formal approach may be assumed that fuzzy data are obtained under a randomness process. Therefore, a framework based on the notion of random fuzzy-rough set may be developed.

Appendix

Here, we present and exemplify some notions of fuzzy logical operators [4], [10], [11], [55] that are triangular norm, triangular conorm, negator, dual, T-residuated implication, and its dual operation.

A triangular norm, or shortly T-norm, is a function $T : [0,1] \times [0,1] \to [0,1]$ that satisfies the following conditions: monotonicity $(T(x,y) \leq T(\alpha,\beta))$; commutativity: $(T(x,y) = T(y,x))$; associativity: $(T(T(x,y),z) = T(x,T(y,z)))$; and boundary condition: $(T(x,1) = x)$. The most popular continuous T-norms include the standard min operator (the largest T-norm) $T_M(x,y) = \min\{x,y\}$; the bounded intersection (also called the Lukasiewicz T-norm) $T_L(x,y) = \max\{0, x+y-1\}$.

A triangular conorm, or shortly T-conorm, is an increasing, commutative, and associative function $S : [0,1] \times [0,1] \to [0,1]$ that satisfies the boundary condition: $\forall x \in [0,1], S(x,0) = x$. The well-known continuous T-conorms include the standard max operator (the smallest T-conorm) $S_M(x,y) = \max\{x,y\}$; the bounded sum $S_L(x,y) = \min\{1, x+y\}$.

A negator N is a decreasing function $N : [0,1] \to [0,1]$ that satisfies $N(0) = 1$ and $N(1) = 0$. A negator N is called involutive iff $N(N(x)) = x$ for all $x \in [0,1]$; it is called weakly involutive iff $N(N(x)) \geq x$ for all $x \in [0,1]$. The standard negator is defined as $N_S(x) = 1 - x$. Given a negator N, T-norm T and T-conorm S are called dual w.r.t. N iff De Morgan laws are satisfied, i.e., $S(N(x),\ N(y)) = N(T(x,y))$ and $T(N(x), N(y)) = N(S(x,y))$.

Given a lower semicontinuous triangular norm T, the residuation implication, or called T-residuated implication is a function $\vartheta : [0,1] \times [0,1] \to [0,1]$ that satisfies $\vartheta(x,y) = sup\{z | z \in [0,1], T(x,z) \leq y\}$ for every $x, y \in [0,1]$. Two examples of T-residuated implications include the Godel implication ϑ_M, based on

$$T_M : \vartheta_M = \begin{cases} 1, & x \leq y \\ y, & x > y \end{cases};$$

the Lukasiewicz implication ϑ_L, based on T_L: $\vartheta_{T_L} = \min\{1 - x + y, 1\}$.

Given a upper semicontinuous triangular conorm S, the dual of $T-$ residuated implication w.r.t. N is a function σ: $[0,1] \times [0,1] \to [0,1]$ that satisfies $\sigma(x,y) = inf\{z \mid z \in [0,1], S(x,z) \geq y\}$ for every $x, y \in [0,1]$. $\sigma(N(x), N(y)) = N(\vartheta(x,y))$ and $\vartheta(N(x), N(y)) = N(\sigma(x,y))$ hold for an involutive negator N. Examples of the dual of T-residuated implication w.r.t. N include the dual of Godel implication σ_M, based on

$$S_M : \sigma_M = \begin{cases} 0, & x \geq y \\ y, & x < y \end{cases};$$

the dual of Lukasiewicz implication σ_L, based on S_L: $\sigma_L = \max\{-x + y, 0\}$.

Acknowledgments

This work was supported in part by the Hong Kong RGC CERG research grants: PolyU 5273/05E (B-Q943) and PolyU 5281/07E (B-Q06C). Chen Degang is supported by the NSFC (70871036). This work is also partially supported by the NSF of Hebei Province (F2008000635, 08963522D, 06213548).

References

[1] Z. Pawlak and A. Skowron, "Rough Sets: Some Extensions," *Information Sciences*, vol. 177, pp. 28-40, 2007.
[2] L.A. Zadeh, "Fuzzy Sets," *Information and Control*, vol. 8, pp. 338-353, 1965.
[3] D. Dubois and H. Prade, "Rough Fuzzy Sets and Fuzzy Rough Sets," *Int'l J. General Systems*, vol. 17, pp. 191-208, 1990.
[4] D. Dubois and H. Prade, "Putting Rough Sets and Fuzzy Sets Together," *Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory*, R. Slowinski, ed., pp. 203-232, Kluwer Academic Publishers, 1992.
[5] J.S. Mi and W.X. Zhang, "An Axiomatic Characterization of a Fuzzy Generalization of Rough Sets," *Information Sciences*, vol. 160, pp. 235-249, 2004.
[6] W.Z. Wu, J.S. Mi, and W.X. Zhang, "Generalized Fuzzy Rough Sets," *Information Sciences*, vol. 151, pp. 263-282, 2003.
[7] W.Z. Wu and W.X. Zhang, "Constructive and Axiomatic Approaches of Fuzzy Approximation Operators," *Information Sciences*, vol. 159, pp. 233-254, 2004.
[8] D.G. Chen, X.Z. Wang, D.S. Yeung, and E.C.C. Tsang, "Rough Approximations on a Complete Completely Distributive Lattice with Applications to Generalized Rough Sets," *Information Sciences*, vol. 176, pp. 1829-1848, 2006.
[9] D.S. Yeung, D.G. Chen, E.C.C. Tsang, J.W.T. Lee, and X.Z. Wang, "On the Generalization of Fuzzy Rough Sets," *IEEE Trans. Fuzzy Systems*, vol. 13, no. 3, pp. 343-361, June 2005.

[10] N.N. Morsi and M.M Yakout, "Axiomatics for Fuzzy Rough Sets," *Fuzzy Sets and Systems,* vol. 100, pp. 327-342, 1998.

[11] A.M. Radzikowska and E.E. Kerre, "A Comparative Study of Fuzzy Rough Sets," *Fuzzy Sets and Systems,* vol. 126, pp. 137-155, 2002.

[12] M.D. Cock, C. Cornelis, and E.E. Kerre, "Fuzzy Rough Sets: The Forgotten Step," *IEEE Trans. Fuzzy Systems,* vol. 15, no. 1, pp. 121-130, Feb. 2007.

[13] C. Cornelis, M.D. Cock, and A.M. Radzikowska, "Fuzzy Rough Sets: From Theory into Practice," *Handbook of Granular Computing,* W. Pedrycz, A. Skowron, and V. Kreinovich, eds., Springer-Verlag, 2008.

[14] R. Slowinski and D. Vanderpooten, "A Generalized Definition of Rough Approximations Based on Similarity," *IEEE Trans. Knowledge and Data Eng.,* vol. 12, no. 2, pp. 331-336, Mar./Apr. 2000.

[15] J.M.F. Salido and S. Murakami, "Rough Set Analysis of a General Type of Fuzzy Data Using Transitive Aggregations of Fuzzy Similarity Relations," *Fuzzy Sets and Systems,* vol. 139, pp. 635-660, 2003.

[16] Y.Y. Yao, "Combination of Rough and Fuzzy Sets Based on Alpha-Level Sets," *Rough Sets and Data Mining: Analysis for Imprecise Data,* T.Y. LIN and N. Cercone, eds., pp. 301-321, Kluwer Academic Publishers, 1997.

[17] Q. Hu, Y. Daren, and X. Zongxia, "Fuzzy Probabilistic Approximation Spaces and Their Information Measures," *IEEE Trans. Fuzzy Systems,* vol. 14, no. 2, pp. 191-201, Apr. 2006.

[18] R. Jensen and Q. Shen, "Fuzzy-Rough Attributes Reduction with Application to Web Categorization," *Fuzzy Sets and Systems,* vol. 141, pp. 469-485, 2004.

[19] Q. Shen and R. Jensen, "Selecting Informative Features with Fuzzy-Rough Sets and Its Application for Complex Systems Monitoring," *Pattern Recognition,* vol. 37, pp. 1351-1363, 2004.

[20] R. Jensen and Q. Shen, "Semantics-Preserving Dimensionality Reduction: Rough and Fuzzy-Rough-Based Approaches," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 12, pp. 1457-1471, Dec. 2004.

[21] R. Jensen and Q. Shen, "Fuzzy-Rough Sets Assisted Attribute Selection," *IEEE Trans. Fuzzy Systems,* vol. 15, no. 1, pp. 73-89, Feb. 2007.

[22] R. Jensen and Q. Shen, "New Approaches to Fuzzy-Rough Feature Selection," *IEEE Trans. Fuzzy Systems,* vol. 17, no. 4, pp. 824-838, Aug. 2009.

[23] E.C.C. Tsang, D.G. Chen, D.S. Yeung, X.Z. Wang, and J.W.T. Lee, "Attribute Reduction Using Fuzzy Rough Sets," *IEEE Trans. Fuzzy Systems,* vol. 16, no. 5, pp. 1130-1141, 2008.

[24] D.G. Chen, E.C.C. Tsang, and S.Y. Zhao, "Attributes Reduction with Fuzzy Rough Sets," *Proc. 2007 IEEE Int'l Conf. Systems, Man, and Cybernetics,* vol. 1, pp. 486-491, 2007.

[25] Q.H. Hu, D.R. Yu, and Z.X. Xie, "Information-Preserving Hybrid Data Reduction Based on Fuzzy-Rough Techniques," *Pattern Recognition Letters,* vol. 27, pp. 414-423, 2006.

[26] Q.H. Hu, D.R. Yu, Z.X. Xie, and X.D. Li, "EROS: Ensemble Rough Subspaces," *Pattern Recognition,* vol. 40, pp. 3728-3739, 2007.

[27] Q.H. Hu, Z.X. Xie, and D.R. Yu, "Hybrid Attribute Reduction Based on a Novel Fuzzy-Rough Model and Information Granulation," *Pattern Recognition,* vol. 40, pp. 3509-3521, 2007.

[28] D. Slezak, "Approximate Reducts in Decision Tables," *Proc. Sixth Int'l Conf. Information Processing and Management of Uncertainty in Knowledge-Based Systems,* pp. 1159-1164, 1996.

[29] D. Slezak, "Approximate Entropy Reducts," *Fundamenta Informaticae,* vol. 53, pp. 365-390, 2002.

[30] H.S. Nguyen and D. Slezak, "Approximate Reducts and Association Rules—Corresponding and Complexity Results," *Proc. Int'l Workshop New Directions in Rough Sets, Data Mining, and Granular-Soft Computing (RSFDGrC '99),* N. Zhong, A. Skowron, and S. Ohsuga, eds., pp. 137-145, 1999.

[31] C. Cornelis, G.H. Martín, R. Jensen, and D. Slezak, "Feature Selection with Fuzzy Decision Reducts," *Proc. Third Int'l Conf. Rough Sets and Knowledge Technology (RSKT '08),* 2008.

[32] E.C.C. Tsang, S.Y. Zhao, and J.W.T. Lee, "Rule Induction Based on Fuzzy Rough Sets," *Proc. 2007 Int'l Conf. Machine Learning and Cybernetics,* vol. 5, pp. 3028-3033, Aug. 2007.

[33] C. Cornelis and R. Jensen, "A Noise-Tolerant Approach to Fuzzy-Rough Feature Selection," *Proc. 17th Int'l Conf. Fuzzy System (FUZZ-IEEE '08),* 2008.

[34] R.B. Bhatt and M. Gopal, "On Fuzzy Rough Sets Approach to Feature Selection," *Pattern Recognition Letters,* vol. 26, pp. 1632-1640, 2005.

[35] Q. Shen and A. Chouchoulas, "A Rough-Fuzzy Approach for Generating Classification Rules Pattern Recognition," *Pattern Recognition,* vol. 35, no. 11, pp. 2425-2438, 2002.

[36] S.K. Pal and P. Mitra, "Case Generation Using Rough Sets with Fuzzy Representation," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 3, pp. 292-300, Mar. 2004.

[37] Y. Li, S.C.K. Shiu, and S.K. Pal, "Combining Feature Reduction and Case Selection in Building CBR Classifiers," *IEEE Trans. Knowledge and Data Eng.,* vol. 18, no. 2, pp. 415-429, Mar. 2006.

[38] Q.H. Hu, D.R. Yu, and Z.X. Xie, "Neighborhood Classifiers," *Expert Systems with Applications,* vol. 34, pp. 866-876, 2008.

[39] G.Q. Cao, S.C.K. Shiu, and X.Z. Wang, "A Fuzzy-Rough Approach for the Maintenance of Distributed Case-Based-Reasoning Systems," *Soft Computing,* vol. 7, no. 8, pp. 491-499, 2003.

[40] M. Sarkar, "Fuzzy-Rough Nearest Neighbor Algorithms in Classification," *Fuzzy Sets and Systems,* vol. 158, no. 19, pp. 2134-2152, 2007.

[41] R.B. Bhatt and M. Gopal, "FRCT: Fuzzy-Rough Classification Trees," *Pattern Analysis and Applications,* vol. 11, no. 1, pp. 73-88, 2008.

[42] R. Yasdi, "Learning Classification Rules from Database in the Context of Knowledge Acquisition and Representation," *IEEE Trans. Knowledge and Data Eng.,* vol. 3, no. 3, pp. 293-306, Sept. 1991.

[43] Y.C. Tsai, C.H. Cheng, and J.R. Chang, "Entropy-Based Fuzzy Rough Classification Approach for Extracting Classification Rules," *Expert Systems with Applications,* vol. 31, no. 2, pp. 436-443, 2006.

[44] E.C.C. Tsang and S.Y. Zhao, "Decision Table Reduction in KDD: Fuzzy Rough Based Approach, Transaction on Rough Sets," *Lecture Notes in Computer Sciences,* vol. 5946, pp. 177-188, 2010.

[45] X.Z. Wang, E.C.C. Tsang, S.Y. Zhao, D.G. Chen, and D.S. Yeung, "Learning Fuzzy Rules from Fuzzy Samples Based on Rough Set Technique," *Information Sciences,* vol. 177, no. 20, pp. 4493-4514, 2007.

[46] S.Y. Zhao, E.C.C. Tsang, and D.G. Chen, "The Model of Fuzzy Variable Precision Rough Sets," *IEEE Trans. Fuzzy Systems,* vol. 17, no. 2, pp. 451-467, Apr. 2009.

[47] A. Skowron and C. Rauszer, "The Discernibility Matrices and Functions in Information Systems," *Intelligent Decision Support—Handbook of Applications and Advances of the Rough Sets Theory,* R. Slowinski, ed., pp. 331-362, Kluwer Academic Publishers, 1992.

[48] Y.Y. Yao, Y. Zhao, and J. Wang, "On Reduct Construction Algorithms," *Proc. First Int'l Conf. Rough Sets and Knowledge Technology (RSKT '06),* pp. 297-304, July 2006.

[49] http://www.ics.uci.edu/~mlearn/MLRepository.html, 2008.

[50] S.R. Safavian and D. Landgrebe, "A Survey of Decision Tree Classifier Methodology," *IEEE Trans. Systems, Man Cybernetics,* vol. 21, no. 3, pp. 660-674, May/June 1991.

[51] Y.F. Yuan and M.J. Shaw, "Introduction of Fuzzy Decision Tree," *Fuzzy Sets and Systems,* vol. 69, pp. 125-139, 1995.

[52] J.R. Quinlan, See5, Release 1.13, http://www.rulequest.com/, 2009.

[53] J.C. Bezdek and J.O. Harris, "Fuzzy Partitions and Relations: An Axiomatic Basis of Clustering," *Fuzzy Sets and Systems,* vol. 84, pp. 143-153, 1996.

[54] T. Sudkamp, "Similarity, Interpolation, and Fuzzy Rule Construction," *Fuzzy Sets and Systems,* vol. 58, pp. 73-86, 1993.

[55] G.J. Klir and B. Yuan, *Fuzzy Logic: Theory and Applications.* Prentice-Hall, 1995.

[56] W. Zhu and F.Y. Wang, "On Three Types of Covering-Based Rough Sets," *IEEE Trans. Knowledge and Data Eng.,* vol. 19, no. 8, pp. 1131-1144, Aug. 2007.

[57] S.H. Nguyen and H.S. Nguyen, "Some Effective Algorithms for Rough Set Methods," *Proc. Conf. Int'l Processing and Management of Uncertainty in Knowledge Based Systems,* pp. 1451-1456, 1996.

Suyun Zhao (M'06) received the bachelor's and master's degrees from the School of Mathematics and Computer Science, Hebei University, Baoding, China, in 2001 and 2004, respectively. She is currently working toward the PhD degree in the Department of Computing, The Hong Kong Polytechnic University. Her research interests are in the areas of machine learning, pattern recognition, fuzzy sets, and rough sets. She is a student member of the IEEE.

Eric C.C. Tsang received the BSc degree in computer studies from the City University of Hong Kong, Kowloon, in 1990, and the PhD degree in computing from The Hong Kong Polytechnic University, in 1996. He is an assistant professor in the Department of Computing, The Hong Kong Polytechnic University. His main research interests are in the areas of fuzzy expert systems, fuzzy neural networks, machine learning, genetic algorithms, and fuzzy support vector machine.

Degang Chen received the MS degree from the Northeast Normal University, Changchun, Jilin, China, in 1994, and the PhD degree from Harbin Institute of Technology, China, in 2000. He has worked as a postdoctoral fellow with Xi'an Jiaotong University, China, from 2000 to 2002 and with Tsinghua University, China, from 2002 to 2004. Since 2006, he has been working as a professor at North China Electric Power University, Beijing, China. His research interests include fuzzy group, fuzzy analysis, rough sets, and SVM.

XiZhao Wang received the PhD degree in computer science from Harbin Institute of Technology, China, in 1998. He is presently the dean and a professor of the College of Mathematics and Computer Science, Hebei University, China. From September 1998 to September 2001, he served as a research fellow in the Department of Computing, The Hong Kong Polytechnic University. He became a full professor and the dean of the College of Mathematics and Computer Science at Hebei University in October 2001. His main research interests include learning from examples with fuzzy representation, fuzzy measures and integrals, neurofuzzy systems and genetic algorithms, feature extraction, multiclassifier fusion, and applications of machine learning. He has more than 160 publications including four books, seven book chapters, and more than 90 journal papers in the *IEEE Transactions on Pattern Analysis and Machine Intelligence*, the *IEEE Transactions on Systems, Man, and Cybernetics*, the *IEEE Transactions on Fuzzy Systems*, *Fuzzy Sets and Systems*, *Pattern Recognition*, etc. He has been the PI/Co-PI for 16 research projects supported partially by the National Natural Science Foundation of China and the Research Grant Committee of Hong Kong Government. He is an IEEE senior member (Board of Governor member in 2005, 2007-2009), the chair of the IEEE SMC Technical Committee on Computational Intelligence, an associate editor of the *IEEE Transactions on SMC, Part B*, an associate editor of the *Pattern Recognition and Artificial Intelligence*, a member of the editorial board of Information Sciences, and an executive member of the Chinese Association of Artificial Intelligence. He was the recipient of the IEEE-SMCS Outstanding Contribution Award in 2004 and the IEEE-SMCS Best Associate Editor Award in 2006. He is the general cochair of the 2002-2009 International Conferences on Machine Learning and Cybernetics, cosponsored by the IEEE SMCS. He is a distinguished lecturer of the IEEE SMC Society.