

Research Article

Adjustable Fuzzy Rough Reduction: A Nested Strategy

Ying Shi,^{1,2} Hui Qi ,¹ Xiaofang Mu,¹ and Mingxing Hou¹

¹School of Computer Science and Technology Department, Taiyuan Normal University, Jinzhong 030619, China

²School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China

Correspondence should be addressed to Hui Qi; qihui@tynu.edu.cn

Received 2 February 2021; Accepted 9 July 2021; Published 23 July 2021

Academic Editor: Daniele Bibbo

Copyright © 2021 Ying Shi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As a crucial extension of Pawlak's rough set, a fuzzy rough set has been successfully applied in real-valued attribute reduction. Nevertheless, the traditional fuzzy rough set is not provided with adjustable ability due to the maximal and minimal operators. It follows that the associated measure for attribute evaluation is not always appropriate. To alleviate such problems, a novel adjustable fuzzy rough set model is presented and further introduced into the parameterized attribute reduction. Additionally, the inner relationship between the appointed parameter and the reduct result is discovered, and thereby a nested mechanism is adopted to accelerate the searching procedure of reduct. Experiments demonstrate that the proposed heuristic algorithm can offer us more stable reducts with higher computational efficiency as compared with the traditional approaches.

1. Introduction

Rough set theory [1] is an effective mathematical tool to qualitatively and quantitatively describe the uncertainty information in data. Due to such characteristics, it has been frequently applied in attribute reduction [2–9], which aims to select a condition attribute subset that can retain the identifiable ability of the original data. It should be pointed out that among existing attribute reduction methods, fuzzy rough set-based ones [10–15] are widely concerned with handling indiscernibility and fuzziness in data with real-valued condition attributes.

Up to now, attribute reduction with the fuzzy rough set has been studied extensively, leading to a large variety of algorithms for computing reducts [16–20]. Actually, the core of these algorithms lies in the fuzzy relation induced from the similarity between every two samples. Generally, a method will be regarded as favorable if it empirically performs well on the attribute evaluation used for identifying the discernibility power of candidate attributes. For example, Hu et al. [21] introduced Gaussian kernels to acquire T-fuzzy equivalence relation and gave a more effective approach for finding reducts; Hu et al. [22] also leveraged k -nearest

neighbor rule to develop a robust attribute reduction method; Ye et al. [23] redefined two types of reflexive fuzzy-neighborhood operators to evaluate attributes appropriately. With a careful reviewing of these methods, unfortunately, these technologies are not provided with adjustable ability. Consequently, no matter how the analytical requirements change, the characterizations of fuzzy rough sets are always fixed. Such a case may result in inflexible attribute evaluation such as the commonly used measure, i.e., approximate quality, which can further cause the undesirable generalization performance of the final reduct.

To fill such a gap, a novel adjustable parameterized fuzzy rough set model is presented firstly, and then the nested strategy is introduced into the process of searching reducts. Experimental results demonstrate that our proposal can offer us more stable reducts with higher computational efficiency as compared with the traditional approaches. The main contributions of this study are outlined as follows:

- (1) An adjustable fuzzy rough set model is constructed. Two special fuzzy relations referred to as strong fuzzy relation and weak fuzzy relation are developed and discussed. Based on these two relations, a

parameterized binary operator is reported and then introduced into the construction of the adjustable fuzzy relation, which can be adjusted by the setting of the parameter. Additionally, we have also suggested some properties of such fuzzy relation.

- (2) The nested mechanism is adopted to perform the procedure of attribute reduction. Our algorithm consists of two aspects. One is attribute evaluation, on the basis of the adjustable fuzzy rough set model, the corresponding approximate quality is presented to measure whether the candidate attribute is qualified or not; the other is searching strategy, considering the appointed parameter in the designed model, the inner relationship between the parameters and the reducts is discovered, and three types of nested reduction are discussed to find the optimal or near optimal reduct, which is defined as forward nested reduction, reverse nested reduction, and weakly nested reduction, respectively.

The rest of this paper is organized as follows. Section 2 provides some background materials on the fuzzy rough set model and its attribute reduction. Section 3 introduces the proposed adjustable fuzzy rough set model, and then the nested strategy-based attribution reduction method is presented. Section 4 describes data sets, evaluation metrics, and experimental settings and then analyzes the results of comparative studies on 6 UCI data sets. Finally, Section 5 summarizes and sets up several issues for future work.

2. Related Work

2.1. Fuzzy Rough Set Model. Formally, a decision system DS can be considered as the 5-tuple $\langle U, C \cup D, V, f \rangle$, in which $U = \{x_1, x_2, \dots, x_n\}$ is a nonempty finite set of n samples called the universe of discourse; $C = \{a_1, a_2, \dots, a_m\}$ is a nonempty finite set of condition attributes aimed to characterize the samples; D is the decision attributes; $\forall a \in C \cup D, V_a$ is the domain of attribute a ; $f: U \times A \rightarrow V$ is an information function.

Let $U \neq \emptyset$ be a universe of discourse. $F: U \rightarrow [0, 1]$ is a fuzzy set [24] on U , $F(x) \in [0, 1]$ is the membership function of F . $F(U)$ is the set of all fuzzy sets on U , fuzzy subset $R \in F(U \times U)$ is a fuzzy relation, and (U, R) is referred to as a fuzzy approximation space. A fuzzy relation R may own the following properties: (1) R is linear if and only if $\forall x \in U, \exists y \in U, R(x, y) = 1$ holds; (2) R is reflexive if and only if $\forall x \in U, R(x, x) = 1$ holds; (3) R is symmetric if and only if $\forall x, y \in U, R(x, y) = R(y, x)$ holds; (4) R is transitive if and only if $\forall x, y, z \in U, \wedge y (R(x, y), R(y, z)) \leq R(x, z)$. The fuzzy relation R discussed in this paper is reflexive at least.

Definition 1. Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall a_t \in C$, the fuzzy relation induced by attribute a_t is represented in the form of the following:

$$R_t = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \dots & \dots & \dots & \dots \\ r_{n1} & r_{n2} & \dots & r_{nn} \end{pmatrix}, \quad (1)$$

where n is the number of samples in U , and $r_{ij} \in [0, 1]$ is the similarity between x_i and x_j .

By the fuzzy relation in Definition 1, we can have the following:

- (1) $R_1 = R_2 \Leftrightarrow \forall x_i, x_j \in U, R_1(x_i, x_j) = R_2(x_i, x_j)$ holds.
- (2) $R = R_1 \cup R_2 \Leftrightarrow \forall x_i, x_j \in U, R = \max\{R_1(x_i, x_j), R_2(x_i, x_j)\}$ holds.
- (3) $R = R_1 \cap R_2 \Leftrightarrow \forall x_i, x_j \in U, R = \min\{R_1(x_i, x_j), R_2(x_i, x_j)\}$ holds.
- (4) $R_1 \subseteq R_2 \Leftrightarrow \forall x_i, x_j \in U, R_1(x_i, x_j) \leq R_2(x_i, x_j)$ holds.

Definition 2 (see [24]). Let $U \neq \emptyset$ be a universe of discourse, R is a fuzzy relation on U , $\forall F \in F(U)$, the fuzzy lower and upper approximations of F in the fuzzy approximation space (U, R) are denoted by $\underline{R}(F)$ and $\overline{R}(F)$, respectively, $\forall x \in U$, the memberships that x belongs to, $\underline{R}(F)$ and $\overline{R}(F)$, are defined as follows:

$$\begin{aligned} \underline{R}(F)(x) &= \inf_{y \in U} \max(1 - R(x, y), F(y)), \\ \overline{R}(F)(x) &= \sup_{y \in U} \min(R(x, y), F(y)), \end{aligned} \quad (2)$$

where $R(x, y)$ is the similarity between x and y based on the fuzzy relation R .

The pair $[\underline{R}(F), \overline{R}(F)]$ is referred to as a fuzzy rough set of F .

2.2. Attribute Reduction in Fuzzy Rough Set. Attribute reduction is an effective way to eliminate irrelevant and redundant information in a decision system. It aims to achieve the minimal subset of condition attributes, which preserves a specified measure of discernibility power invariant. The common measures of discernibility power contain dependency function [25–27], information entropy [28, 29], and monotonic measure [30]. For simplicity, we take approximate quality as an example in this subsection.

Definition 3. Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall B \subseteq C, R_B$ is the fuzzy relation generated by the set of attributes B , $U|IND(D) = \{d_1, d_2, \dots, d_p\}$ is the partition induced by the decision D , and then the approximate quality of $U|IND(D)$ with respect to B in the fuzzy rough set is defined as follows:

$$\gamma(B, D) = \frac{|U_{i=1}^p \underline{R}_B(d_i)|}{|U|} = \frac{\sum_{j=1}^{|U|} \max_{1 \leq i \leq p} (R_B(d_i)(x_j))}{|U|}, \quad (3)$$

where U is the sample set, B is the attribute set, D is the decision set, R_B is the fuzzy relation generated by the set of attributes B , x_j is the j th sample combined with U , $|\bullet|$ is the cardinality of a set, and d_i is a decision class.

By Definition 3, we can see that $\gamma(B, D)$ reflects the approximation abilities of the granulated spaces generated by the set of attributes B to characterize the decision D . Literatures [21, 31] have proved that approximate quality is monotonic with the increasing or decreasing of condition attributes in a decision system, i.e., $\gamma(B_1, D) \leq \gamma(B_2, D)$ if $B_1 \subseteq B_2 \subseteq C$. In addition, it is obvious that $0 \leq \gamma(B, D) \leq 1$ holds.

Definition 4. Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall B \subseteq C$, B is referred to as a reduct of C if and only if

- (1) $\gamma(B, D) \neq \gamma(C, D)$
- (2) $\forall B' \subset B, \gamma(B', D) \neq \gamma(B, D)$

In a fuzzy rough set, the similarity of samples will decrease with the number of attributes increasing, and then the lower approximation of the decision system will increase. That is to say, the positive region will be enlarged. It is well known that the samples of the positive region are usually regarded as to be certain. Therefore, the certainty of the decision system will improve. It is consistent with our intuition that new attributes (features) will bring more information about granulation and classification.

Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall a_t \in B \subseteq C$, and we define a coefficient as the significance of a_t relative to decision D .

$$\text{Sig}_{\text{in}}(a_t, B, D) = \gamma(B, D) - \gamma(B - \{a_t\}, D), \quad (4)$$

$\text{Sig}_{\text{in}}(a_t, B, D)$ reflects the change of approximate quality if attribute a_t is eliminated from B . Similarly, we can also define the following:

$$\text{Sig}_{\text{out}}(a_t, B, D) = \gamma(B + \{a_t\}, D) - \gamma(B, D), \quad (5)$$

where $a_t \in C - B$, $\text{Sig}_{\text{out}}(a_t, B, D)$ reflects the change of approximate quality if attribute a_t is introduced into B . From the above perspective, many researchers [9, 32–34] iteratively select the most significant attributes with forward greedy algorithm until no more deterministic rules generating with the increasing of attributes.

Formally, a forward attribute reduction algorithm can be designed as follows.

For Algorithm 1, if approximate quality $\gamma(\cdot, \cdot)$ is replaced by some other measures of discernibility power, then it can also be used to compute reducts with different measurement indices.

3. Adjustable Fuzzy Rough Set

3.1. Strong and Weak Fuzzy Rough Set Model. In $\langle U, C \cup D, V, f \rangle$, $\forall a_t \in C$ ($1 \leq t \leq m$), a series of fuzzy relations R_t can be induced by each attribute a_t . For all the m condition attributes, we can consider the following two special fuzzy relations.

- (1) Strong fuzzy relation: $R^S = R_1 \cap R_2 \cap \dots \cap R_m$, R^S corresponds to the strong fuzzy relation matrix $M(R^S)$. In $M(R^S)$, $\forall x_i, x_j \in U$, the similarity between x_i and x_j , denoted by r_{ij}^S , is minimum.
- (2) Weak fuzzy relation: $R^W = R_1 \cup R_2 \cup \dots \cup R_m$, R^W corresponds to the weak fuzzy relation matrix $M(R^W)$. In $M(R^W)$, $\forall x_i, x_j \in U$, the similarity between x_i and x_j , denoted by r_{ij}^W , is maximum.

Definition 5. Let $U \neq \emptyset$ be a universe of discourse, R^S is a strong fuzzy relation on U , $\forall F \in F(U)$, the strong fuzzy lower and upper approximations of F in the fuzzy approximation space (U, R^S) are denoted by $\underline{R}^S(F)$ and $\overline{R}^S(F)$, respectively, $\forall x \in U$, the membership that x belongs to $\underline{R}^S(F)$ and $\overline{R}^S(F)$ are defined as follows:

$$\begin{aligned} \underline{R}^S(F)(x) &= \inf_{y \in U} \max(1 - R^S(x, y), F(y)), \\ \overline{R}^S(F)(x) &= \sup_{y \in U} \min(R^S(x, y), F(y)), \end{aligned} \quad (6)$$

where $R^S(x, y)$ is the similarity between x and y based on the strong fuzzy relation R^S .

The pair $[\underline{R}^S(F), \overline{R}^S(F)]$ is referred to as a strong fuzzy rough set of F .

Definition 6. Let $U \neq \emptyset$ be a universe of discourse, R^W is a weak fuzzy relation on U , $\forall F \in F(U)$, and the weak fuzzy lower and upper approximations of F in the fuzzy approximation space (U, R^W) are denoted by $\underline{R}^W(F)$ and $\overline{R}^W(F)$, respectively, $\forall x \in U$, the memberships that x belongs to, $\underline{R}^W(F)$ and $\overline{R}^W(F)$, are defined as follows:

$$\begin{aligned} \underline{R}^W(F)(x) &= \inf_{y \in U} \max(1 - R^W(x, y), F(y)), \\ \overline{R}^W(F)(x) &= \sup_{y \in U} \min(R^W(x, y), F(y)), \end{aligned} \quad (7)$$

where $R^W(x, y)$ is the similarity between x and y based on the weak fuzzy relation R^W .

The pair $[\underline{R}^W(F), \overline{R}^W(F)]$ is referred to as a weak fuzzy rough set of F .

3.2. Adjustable Fuzzy Rough Set Model. In a general way, suppose that M is an index set, $\forall t \in M$, $\forall kt \in [0, 1]$, then a parameterized binary operator [17] can be defined as follows:

$$\lambda \underset{t \in M}{\vee} k_t = \begin{cases} \max_{t \in M} k_t: \{k_t: t \in M\} \subseteq [0, \lambda], \\ \min_{t \in M} k_t: \{k_t: t \in M\} \subseteq [\lambda, 1], \\ \lambda: \text{others,} \end{cases} \quad (8)$$

where $\lambda \in [0, 1]$. Especially, let index set $M = 2$, $\forall k_1, k_2 \in [0, 1]$, then

$$k_1 \lambda \underset{2}{\vee} k_2 = \begin{cases} \max(k_1, k_2): k_1, k_2 \in [0, \lambda], \\ \min(k_1, k_2): k_1, k_2 \in [\lambda, 1], \\ \lambda: \text{others,} \end{cases} \quad (9)$$

(i) **Inputs:** A decision system $DS = \langle U, C \cup D, V, f \rangle$
(ii) **Outputs:** A reduct B .
(iii) **Step 1.** $B \leftarrow \emptyset$, compute $\gamma(C, D)$;
Step 2. $\forall a_t \in C$, compute $\text{Sig}_{\text{in}}(a_t, C, D) = \gamma(C, D) - \gamma(C - \{a_t\}, D)$
Step 3., where $\text{Sig}_{\text{in}}(a_t, C, D) = \max\{\text{Sig}_{\text{in}}(a_t, C, D) : \forall a_t \in C\}$
Step 4. Compute $\gamma(B, D)$
(iv) **Step 5. Do**
(1) compute $\forall a_t \in C - B$ $\text{Sig}_{\text{out}}(a_t, B, D) = \gamma(B + \{a_t\}, D) - \gamma(B, D)$
(2) $B \leftarrow B \cup \{a_t\}$ where $\text{Sig}_{\text{out}}(a_t, B, D) = \max\{\text{Sig}_{\text{out}}(a_t, B, D) : \forall a_t \in C - B\}$
(3) Compute $\gamma(B, D)$;
Until $\gamma(B, D) = \gamma(C, D)$
Step 6. $\forall a_t \in B$, **if** $\gamma(B - \{a_t\}, D) = \gamma(C, D)$ **then** $B \leftarrow B - \{a_t\}$

ALGORITHM 1: Heuristic approach to compute reduct (HACR).

where $\lambda \in [0, 1]$. Obviously, λ is a generalization of maximal operator (max) and minimal operator (min), $\bigvee_{t \in M} k_t$ can be adjusted by setting up the value of λ . Especially, if $\lambda = 0$, then $\bigvee = \min$; if $\lambda = 1$, then $\bigvee = \max$.

Theorem 1. Let M is an index set, $\forall t \in M, \forall k_t \in [0, 1]$, then $\min_{t \in M} k_t \leq \bigvee_{t \in M} k_t \leq \max_{t \in M} k_t$.

Proof. By formula (11), we can discuss in following three cases:

- (1) If $\{k_t : t \in M\} \subseteq [0, \lambda]$, then $\min_{t \in M} k_t \leq \bigvee_{t \in M} k_t = \max_{t \in M} k_t$ holds.
- (2) If $\{k_t : t \in M\} \subseteq [\lambda, 1]$, then $\min_{t \in M} k_t = \bigvee_{t \in M} k_t \leq \max_{t \in M} k_t$ holds.
- (3) Otherwise, it is obvious that $\min_{t \in M} k_t \leq \bigvee_{t \in M} k_t = \lambda \leq \max_{t \in M} k_t$ holds.

In $\langle U, C \cup D, V, f \rangle$, $\forall a_t \in C (1 \leq t \leq m)$, a series of fuzzy relations R_t can be induced by each attribute at $\forall x_i, x_j \in U$, the similarity between x_i and x_j based on all the m fuzzy relations, denoted by r_{ij}^λ , can be adjusted by the above binary operator. The fuzzy relation constructed by r_{ij}^λ is referred to as an adjustable fuzzy relation, and it is denoted by R^λ . R^λ corresponds to the adjustable fuzzy relation matrix $M(R^\lambda)$. \square

Theorem 2. Let $\langle U, C \cup D, V, f \rangle$ be a decision system. The adjustable fuzzy relation R^λ can be obtained by adjusting the value of λ between strong fuzzy relation R^s and weak fuzzy relation R^w , namely, $\forall x_i, x_j \in U$, then $r_{ij}^\lambda = r_{ij}^s \bigvee r_{ij}^w$.

Proof. Since $r_{ij}^s \leq r_{ij}^w$, we can discuss in the following three cases:

- (1) $\forall x_i, x_j \in U$, If $\{r_{ij}^s, r_{ij}^w\} \subseteq [0, \lambda]$, by formula (8), then $r_{ij}^\lambda = \bigvee_{1 \leq t \leq m} r_{ij}^t = \max_{1 \leq t \leq m} r_{ij}^t = r_{ij}^w$ ($-b \pm \sqrt{b^2 - 4ac}/2a$), by formula (9), then $r_{ij}^s \bigvee r_{ij}^w = \max(r_{ij}^s, r_{ij}^w) = r_{ij}^w$, accordingly, $r_{ij}^\lambda = r_{ij}^s \bigvee r_{ij}^w$ holds.

- (2) $\forall x_i, x_j \in U$, If $\{r_{ij}^s, r_{ij}^w\} \subseteq [\lambda, 1]$, by formula (8), then $r_{ij}^\lambda = \bigvee_{1 \leq t \leq m} r_{ij}^t = \min_{1 \leq t \leq m} r_{ij}^t = r_{ij}^s$, by formula (9), then $r_{ij}^s \bigvee r_{ij}^w = \min(r_{ij}^s, r_{ij}^w) = r_{ij}^s$, accordingly, $r_{ij}^\lambda = r_{ij}^s \bigvee r_{ij}^w$ holds.
- (3) Otherwise, by formula (8), $r_{ij}^\lambda = \bigvee_{1 \leq t \leq m} r_{ij}^t = \lambda$, by formula (9), $r_{ij}^s \bigvee r_{ij}^w = \lambda$, accordingly, $r_{ij}^\lambda = r_{ij}^s \bigvee r_{ij}^w$ holds. \square

Definition 7. Let $U \neq \emptyset$ be a universe of discourse, R^λ is an adjustable fuzzy relation on U , $\forall F \in F(U)$, the adjustable fuzzy lower and upper approximations of F in the fuzzy approximation space (U, R^λ) are denoted by $\underline{R}^\lambda(F)$ and $\overline{R}^\lambda(F)$, respectively, $\forall x \in U$, the membership that x belongs to $\underline{R}^\lambda(F)$ and $\overline{R}^\lambda(F)$ are defined as follows:

$$\underline{R}^\lambda(F)(x) = \inf_{y \in U} \max(1 - R^\lambda(x, y), F(y)), \quad (10)$$

$$\overline{R}^\lambda(F)(x) = \sup_{y \in U} \min(R^\lambda(x, y), F(y)), \quad (11)$$

where $R^\lambda(x, y)$ is the similarity between x and y based on the adjustable fuzzy relation R^λ .

The pair $[\underline{R}^\lambda(F), \overline{R}^\lambda(F)]$ is referred to as an adjustable fuzzy rough set of F .

Theorem 3. Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall F \in F(U)$, then

- (1) $\underline{R}^0(F) = \underline{R}^s(F)$;
- (2) $\overline{R}^0(F) = \overline{R}^s(F)$;
- (3) $\underline{R}^1(F) = \underline{R}^w(F)$;
- (4) $\overline{R}^1(F) = \overline{R}^w(F)$.

Proof. By formula (8), we have $\bigvee = \min$ and $\bigvee = \max$.

- (1) By formula (10), $\forall x \in U, \underline{R}^0(F)(x) = \inf_{y \in U} \max(1 - R^0(x, y), F(y)) = \inf_{y \in U} \max(1 - R^s(x, y), F(y)) = \underline{R}^s(F)(x)$, accordingly, $\underline{R}^0(F) = \underline{R}^s(F)$ holds.
- (2) By formula (11), $\forall x \in U, \overline{R}^0(F)(x) = \sup_{y \in U} \min(R^0(x, y), F(y)) = \sup_{y \in U} \min$

- $\frac{(R^s(x, y), F(y))}{R^0(F)} = \overline{R^s}(F)(x)$, accordingly,
 $R^0(F) = \overline{R^s}(F)$ holds.
- (3) By formula (10), $\forall x \in U$, $\frac{R^1(F)(x)}{R^1(x, y), F(y)} = \inf_{y \in U} \max(1 - R^w(x, y), F(y)) = \inf_{y \in U} \max(1 - R^w(x, y), F(y)) = \frac{R^w(F)(x)}{R^1(F)}$, accordingly,
 $\frac{R^1(F)}{R^1(F)} = \frac{R^w(F)}{R^1(F)}$ holds.
- (4) By formula (11), $\forall x \in U$, $\frac{\overline{R^1}(F)(x)}{\sup_{y \in U} \min(R^1(x, y), F(y))} = \sup_{y \in U} \min(R^w(x, y), F(y)) = \sup_{y \in U} \min(R^w(x, y), F(y)) = \frac{\overline{R^w}(F)(x)}{R^1(F)}$, accordingly,
 $\frac{R^1(F)}{R^1(F)} = \frac{\overline{R^w}(F)}{R^1(F)}$ holds.

Theorem 3 tells us that both strong and weak fuzzy rough sets are special cases of an adjustable fuzzy rough set, and an adjustable fuzzy rough set is a generalization of strong and weak fuzzy rough sets. \square

Theorem 4. Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall F \in F(U)$, then

- (1) $\frac{R^w(F)}{R^w(F)} \subseteq \frac{R^\lambda(F)}{R^\lambda(F)} \subseteq \frac{R^s(F)}{R^s(F)}$;
- (2) $\frac{\overline{R^s}(F)}{\overline{R^s}(F)} \subseteq \frac{\overline{R^\lambda}(F)}{\overline{R^\lambda}(F)} \subseteq \frac{\overline{R^w}(F)}{\overline{R^w}(F)}$.

Proof. By Theorem 1, we have $\min_{t \in M} k_t \leq \bigvee_{t \in M} k_t \leq \max_{t \in M} k_t$, then $R^S(x, y) \leq R^\lambda(x, y) \leq R^w(x, y)$.

- (1) $\forall x \in \frac{R^w(F)}{R^w(F)}$, $\frac{R^w(F)(x)}{R^w(x, y), F(y)} = \inf_{y \in U} \max(1 - R^w(x, y), F(y)) \leq \inf_{y \in U} \max(1 - R^\lambda(x, y), F(y)) = \frac{R^\lambda(F)(x)}{R^\lambda(F)}$, accordingly, $\frac{R^w(F)}{R^w(F)} \subseteq \frac{R^\lambda(F)}{R^\lambda(F)}$ holds. Similarly, it is not difficult to prove $\frac{R^\lambda(F)}{R^\lambda(F)} \subseteq \frac{R^s(F)}{R^s(F)}$ holds.
- (2) $\forall x \in \frac{\overline{R^s}(F)}{\overline{R^s}(F)}$, $\frac{\overline{R^s}(F)(x)}{\sup_{y \in U} \min(R^s(x, y), F(y))} \leq \sup_{y \in U} \min(R^\lambda(x, y), F(y)) = \frac{\overline{R^\lambda}(F)(x)}{\overline{R^\lambda}(F)}$, accordingly, $\frac{\overline{R^s}(F)}{\overline{R^s}(F)} \subseteq \frac{\overline{R^\lambda}(F)}{\overline{R^\lambda}(F)}$ holds. Similarly, it is not difficult to prove $\frac{\overline{R^\lambda}(F)}{\overline{R^\lambda}(F)} \subseteq \frac{\overline{R^w}(F)}{\overline{R^w}(F)}$ holds.

Theorem 4 tells us that adjustable fuzzy lower approximation is between weak and strong fuzzy lower approximations, and adjustable fuzzy upper approximation is between strong and weak fuzzy upper approximations. \square

3.3. Approximate Reduction in Adjustable Fuzzy Rough Set. As discussed in Subsection 2.2, approximate quality reflects the relevance between condition attributes and decision, and it can be used to measure the significance of a candidate attribute. Similarly, we will give the definition of approximate quality in strong, weak, and adjustable fuzzy rough sets firstly.

Definition 8. Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall B \subseteq C$, R_B^s, R_B^w and R_B^λ are the strong, weak, and adjustable fuzzy relations generated by the set of attributes B , respectively, $U|IND(D) = \{d_1, d_2, \dots, d_p\}$ is the partition induced by the decision D , then the approximate quality of $U|IND(D)$ with respect to B in the strong, weak, and adjustable fuzzy rough sets are defined as follows, respectively:

$$\gamma^s(B, D) = \frac{|U_{i=1}^p R_B^s(d_i)|}{|U|} = \frac{\sum_{j=1}^{|U|} \max_{1 \leq i \leq p} (R_B^s(d_i)(x_j))}{|U|},$$

$$\gamma^w(B, D) = \frac{|U_{i=1}^p R_B^w(d_i)|}{|U|} = \frac{\sum_{j=1}^{|U|} \max_{1 \leq i \leq p} (R_B^w(d_i)(x_j))}{|U|},$$

$$\gamma^\lambda(B, D) = \frac{|U_{i=1}^p R_B^\lambda(d_i)|}{|U|} = \frac{\sum_{j=1}^{|U|} \max_{1 \leq i \leq p} (R_B^\lambda(d_i)(x_j))}{|U|}, \quad (12)$$

where R_B^s, R_B^w , and R_B^λ are the strong, weak, and adjustable fuzzy relations generated by the set of attributes B , respectively, U is the sample set, B is the attribute set, D is the decision set, $|\cdot|$ is the cardinality of a set, and d_i is a decision class.

Theorem 5. Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall B \subseteq C$, then $\gamma^w(B, D) \leq \gamma^\lambda(B, D) \leq \gamma^s(B, D)$.

Proof. By Theorem 4, we have $\frac{R_B^w(d_i)}{R_B^w(d_i)} \subseteq \frac{R_B^\lambda(d_i)}{R_B^\lambda(d_i)} \subseteq \frac{R_B^s(d_i)}{R_B^s(d_i)}$, it is obvious that $\gamma^w(B, D) \leq \gamma^\lambda(B, D) \leq \gamma^s(B, D)$ holds.

Theorem 5 tells us that approximation quality in the adjustable fuzzy rough set is between that in weak and strong fuzzy rough sets.

Algorithm 1 can also be applicable for attribute reduction in the adjustable fuzzy rough set, which is similar to the classic fuzzy rough set. Nevertheless, in practice, the criteria of reduction are too harsh in Algorithm 1. To expand the application scope of attribute reduction (dimension reduction, feature selection), some researchers [9, 21, 27, 28, 31, 35] have introduced the threshold ε to control the change of discernibility power for relaxing the criteria of reduction, and they also consider B as a reduct of C satisfying the following conditions: (1) $\gamma(C, D) - \gamma(B, D) \leq \varepsilon$; (2) $\forall B' \subset B$, $\gamma(C, D) - \gamma(B', D) > \varepsilon$. The above conditions show that ε is aimed at eliminating redundant attributes as much as possible while maintaining the change of approximate quality in a smaller range. In general, ε is recommended to be 0%–10% of the original approximate quality in a decision system.

Formally, a tolerant forward approximate reduction algorithm in the adjustable fuzzy rough set can be designed as follows. \square

3.4. Nested Strategy-Based Reduction in Adjustable Fuzzy Rough Set. Algorithm 2 is an enhancement version of Algorithm 1, which introduced a tolerant parameter to improve the efficiency of attribute reduction. However, in real-world applications, we always encounter data with too many objects or attributes. In this circumstance, starting with an empty set for the attribute reduction process may be time consuming and result in algorithm performance degradation. In order to deal with this issue, a nested strategy-based attribute reduction approach is introduced into the

adjustable fuzzy rough sets model. In this subsection, some notions of the nested strategy-based reduction, i.e., Forward Nested Reduction, Reverse Nested Reduction, and Weakly Nested Reduction, are firstly presented.

Definition 9 (Forward Nested Reduction [36]). In a parameterized model, if the reducts on different approximations satisfy $\forall \text{Reduct}(\beta), \exists \text{Reduct}(\alpha)$, we have $\text{Reduct}(\alpha) \subseteq \text{Reduct}(\beta)$ which holds; then this type of nested structure is “Forward Nested Reduction.”

Definition 10 (Reverse Nested Reduction [36]). In a parameterized model, if the reducts on different approximations satisfy $\forall \text{Reduct}(\beta), \exists \text{Reduct}(\alpha)$, we have $\text{Reduct}(\alpha) \supseteq \text{Reduct}(\beta)$ which holds; then this type of nested structure is “Reverse Nested Reduction.”

Definition 11 (Weakly Nested Reduction [36]). In a parameterized model, if the reducts on different approximations satisfy $\forall \text{Reduct}(\beta), \exists P \supseteq \text{Reduct}(\beta)$, we have $P \supseteq \text{Reduct}(\alpha)$ which holds; then this type of nested structure is “Forward Weakly Nested Reduction.”

By Definition 9–Definition 11, we know that the forward nested structure satisfies that there exists a smaller reduct included in the known larger reduct (the former is a subset of the latter), and the reverse nested structure satisfies that there exists a larger reduct containing the known smaller reduct (the latter is a subset of the former). However, the weakly nested structure is a much weaker relation than the inclusion relation of forward and reverse nested structure, that is, the intersection relation of two reducts.

Theorem 6. *Let $\langle U, C \cup D, V, f \rangle$ be a decision system, $\forall B \subseteq C$, approximate quality $\gamma^\lambda(B, D)$ is monotonic with B in the adjustable fuzzy rough set, then with the parameter λ increasing, the weakly nested reduction holds.*

Theorem 6 presents a necessary condition for constructing nested attribute reduction. In light of the nested strategy of reduction, an algorithm is designed as follows to search for a different reduct quickly with an existing reduct. The detailed processes are presented in Algorithm 3.

The most striking difference between Algorithm 3 and Algorithm 2 is that the prereduct is considered for searching new reduct in Algorithm 3. When the parameters are changed, the reduction under the new parameters can be obtained by changing only a few attributes sets of reduction with the original parameters. Since the proposed algorithm NS-THACR does not need to add attributes from an empty attribute set, the operation steps of approximation and approximation quality under repeated calculation are greatly reduced, and the efficiency of reduction execution is improved.

4. Experimental Analysis

To evaluate the performance of our nested strategy-based heuristic attribute reduction approach, 6 real-world UCI data sets have been employed in this paper. Table 1 summarizes some details of these experimental data sets. All the

experiments are run on a workstation equipped with a 3.10 GHz processor and a 8.00 G memory. The programming language is Matlab R2014b.

In the context of this paper, 10-fold cross-validation (10-CV) is used for evaluating the effectiveness of the proposed algorithm in our experiments. In the following experiments, all samples are broken into ten groups of the same size, the nine groups compose the training set, and the one group composes the testing set. The attribute reduction process is repeated 10 times in turn and the mean value and standard deviation of 10 experimental results are recorded.

In this section, we focus on the elapsed time comparison for computing reduct by using the nested reduction strategy or not. The comparison in this part is set up as follows: (1) Ten values of the parameter λ are selected on the interval $[0, 1]$ with the step length 0.1. (2) Seven values of ε are chosen to control the changes of approximate quality, i.e., 0%, 2%, 4%, 6%, 8%, 10%. In fact, $\varepsilon = 0\%$ means that the approximate quality of obtained reduct is equal to the original decision system and similar explanations can be achieved for another six values. (3) In order to show the effectiveness of the nested reduction strategy, THACR discussed in Algorithm 2 is presented as a benchmark algorithm. The detailed results are shown in the following Tables 2–7. In these six tables, the first row is the elapsed time of computing THACR and the second row is the elapsed time of computing NS-THACR and the minimum elapsed time is bold. With a careful study of these tables, we can obtain the following remarks.

- (1) From the results in Tables 2–7, it can be seen that for all experiments, the elapsed time of computing NS-THACR is better than the elapsed time of computing THACR, and the calculation time is reduced by more than 80 percent. In addition, the difference between ε and λ also has a certain impact on the experimental results, and the time consumed by the two methods is also different. The choice of these two parameters also requires different attempts in the experiment to search for an appropriate subset of attributes.
- (2) The elapsed time of computing reduct by THACR is higher than that of computing reduct by NS-THACR. For example, when we search for a reduct for dataset *E. coli* (Table 3), if we set $\lambda = 0.3$ and $\varepsilon = 8\%$, then we can note from Table 3 that the average elapsed time of THACR is 2.5724, whereas the average elapsed time of NS-THACR is 0.2282, the computing time is decreased by 91.12 percent. For dataset Seeds (Table 6), in the attribute reduction process, if we choose $\lambda = 0.7$ and $\varepsilon = 2\%$, then we can obtain the average elapsed time of THACR, which is 0.7392 and the average elapsed time of NS-THACR, which is 0.1376, and the computing time, which is decreased by 81.38 percent.
- (3) In general, the elapsed time of $\varepsilon = 0$ is higher than others in these two algorithms. For instance, in dataset Glass (Table 4), if we set $\lambda = 0.8$, we can note that the average elapsed time of THACR is 2.0171 when $\varepsilon = 0\%$, whereas the average elapsed time of

Inputs: A decision system $DS = \langle U, C \cup D, V, f \rangle$, the adjustable parameter λ , the threshold ε for controlling the change of approximate quality;
Outputs: A reduct B .
Step 1. compute $\gamma^\lambda(C, D)$;
Step 2. compute $\text{Sig}_{in}(a_t, C, D) = \gamma^\lambda(C, D) - \gamma^\lambda(C - \{a_t\}, D)$
Step 3. where $\text{Sig}_{in}(a_t, C, D) = \max\{\text{Sig}_{in}(a_t, C, D) : \forall a_t \in C\}$
Step 4. Compute $\gamma^\lambda(B, D)$
Step 5. Do
 (1) compute; $\forall a_t \in C - B$ $\text{Sig}_{out}(a_t, B, D) = \gamma^\lambda(B + \{a_t\}, D) - \gamma^\lambda(B, D)$
 (2) where; $B \leftarrow B \cup \{a_t\}$, Where $\text{Sig}_{out}(a_t, B, D) = \max\{\text{Sig}_{out}(a_t, B, D) : \forall a_t \in C - B\}$
 (3) Compute; $\gamma^\lambda(B, D)$
 Until $\gamma^\lambda(C, D) - \gamma^\lambda(B, D) \leq \varepsilon$
Step 6. $\forall a_t \in B$, if $\gamma^\lambda(C, D) - \gamma^\lambda(B - \{a_t\}, D) \leq \varepsilon$ then $B \leftarrow B - \{a_t\}$

ALGORITHM 2: Tolerant heuristic approach to compute reduct (THACR).

Inputs: A decision system. $DS = \langle U, \text{Reduct}(\lambda_\beta) \cup D, V, f \rangle$ the existing parameter λ_β and a new parameter λ_α , the threshold ε for controlling the change of approximate quality;
Outputs: A new reduct B of λ_α .
Step 1. $B \leftarrow \text{Reduct}(\lambda_\beta)$
Step 2. Compute the approximate qualities of $\gamma^{\lambda_\alpha}(B, D)$ and $\gamma^{\lambda_\alpha}(C, D)$, respectively;
Step 3. If $\gamma^{\lambda_\alpha}(C, D) - \gamma^{\lambda_\alpha}(B, D) > \varepsilon$
 go to **Step 4**
 Else If $\gamma^{\lambda_\alpha}(C, D) - \gamma^{\lambda_\alpha}(B, D) \leq \varepsilon$
 go to **Step 5**
 End
 End
Step 4. Do
 (1) $\forall a_t \in C - B$, compute; $\text{Sig}_{out}(a_t, B, D) = \gamma^{\lambda_\alpha}(B + \{a_t\}, D) - \gamma^{\lambda_\alpha}(B, D)$
 (2) where. $B \leftarrow B \cup \{a_t\}$; $\text{Sig}_{out}(a_t, B, D) = \max\{\text{Sig}_{out}(a_t, B, D) : \forall a_t \in C - B\}$;
 (3) Compute $\gamma^{\lambda_\alpha}(B, D)$;
 Until $\gamma^{\lambda_\alpha}(C, D) - \gamma^{\lambda_\alpha}(B, D) \leq \varepsilon$
Step 5. $\forall a_t \in B$, if $\gamma^{\lambda_\alpha}(C, D) - \gamma^{\lambda_\alpha}(B - \{a_t\}, D) \leq \varepsilon$ then $B \leftarrow B - \{a_t\}$

ALGORITHM 3: Nested strategy-based tolerant heuristic approach to compute reduct (NS-THACR).

TABLE 1: The characteristics of experimental data sets from UCI.

ID	Data sets	Abbr.	Samples	Attributes	Abbr.	Classes
1	Diabetes	Diab	768	8	Real value	2
2	<i>E. coli</i>	<i>E. coli</i>	336	7	Real value	8
3	Glass	Glass	214	10	Real value	6
4	Iris	Iris	150	4	Real value	3
5	Seeds	Seeds	210	7	Real value	3
6	Wine	Wine	178	13	Real value	3

THACR is 1.7782 when $\varepsilon = 6\%$. As far as NS-THACR is concerned, the average elapsed time is 0.2985 when $\varepsilon = 0\%$, whereas the average elapsed time is 0.2032 when $\varepsilon = 6\%$. It is because that $\varepsilon = 0\%$ is too strict; that is, the approximate quality of reduct must be equal to the original dataset. Therefore, more time is needed to search for an appropriate attribute subset.

In addition, the proposed method is compared with two other popular attribute reduction methods, one is the classical Nested Strategy-based Heuristic Approach to Computing Reduct (NS-HACR), and another is DMF-FN [36, 37]. MF-FN is the nested reduction approaches designed by the discernibility matrix. The detailed results of the above six data sets are shown in Figure 1, in which

TABLE 2: Computational time of attribute reduction with using nested reduction or not (mean \pm std. deviation) on diabetes.

Diabetes	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.2$	$\lambda = 0.3$	$\lambda = 0.4$	$\lambda = 0.5$	$\lambda = 0.6$	$\lambda = 0.7$	$\lambda = 0.8$	$\lambda = 0.9$	$\lambda = 1.0$
$\varepsilon = 10\%$	4.5692 \pm 0.1703 1.2313 \pm 0.0703	4.7531 \pm 0.1621 1.0216 \pm 0.0812	4.3546 \pm 0.1478 0.9841 \pm 0.1031	4.5263 \pm 0.1861 0.8791 \pm 0.1131	4.4215 \pm 0.1878 0.9941 \pm 0.0861	4.1463 \pm 0.1527 0.9833 \pm 0.0878	5.1201 \pm 0.1638 0.9167 \pm 0.0527	5.1879 \pm 0.1024 1.1023 \pm 0.0978	4.9872 \pm 0.1632 1.0578 \pm 0.1021	4.7832 \pm 0.1843 0.9221 \pm 0.0632	4.6547 \pm 0.1645 0.9489 \pm 0.0843
$\varepsilon = 8\%$	4.0167 \pm 0.1527 1.0167 \pm 0.0437	4.2361 \pm 0.1226 1.1021 \pm 0.0547	5.0167 \pm 0.0989 0.9167 \pm 0.0842	5.1668 \pm 0.1032 0.9915 \pm 0.0321	5.0333 \pm 0.0703 0.9386 \pm 0.0703	4.9354 \pm 0.0703 0.9386 \pm 0.0703	4.7656 \pm 0.0861 0.9667 \pm 0.0861	4.8972 \pm 0.1098 0.9988 \pm 0.0765	4.9951 \pm 0.0975 0.9745 \pm 0.0823	5.1021 \pm 0.0632 0.9288 \pm 0.0632	4.9941 \pm 0.1078 0.9476 \pm 0.0843
$\varepsilon = 6\%$	4.0310 \pm 0.0655 1.0310 \pm 0.0655	4.1278 \pm 0.0742 1.1573 \pm 0.0462	4.6167 \pm 0.0527 0.9168 \pm 0.0527	5.0132 \pm 0.0714 0.9962 \pm 0.0345	5.0021 \pm 0.1204 0.9987 \pm 0.0012	4.9120 \pm 0.0698 1.0167 \pm 0.0583	4.9756 \pm 0.0671 0.9998 \pm 0.0641	5.0667 \pm 0.0861 1.0667 \pm 0.0861	4.9756 \pm 0.0671 1.1271 \pm 0.0875	5.0101 \pm 0.0752 0.9942 \pm 0.0546	4.9982 \pm 0.0012 1.0400 \pm 0.0343
$\varepsilon = 4\%$	4.9286 \pm 0.0602 1.0286 \pm 0.0602	4.8352 \pm 0.0875 1.0542 \pm 0.0346	5.0286 \pm 0.0602 0.9286 \pm 0.0402	5.1021 \pm 0.0690 0.9578 \pm 0.0321	4.9875 \pm 0.0879 0.9745 \pm 0.0578	4.8562 \pm 0.0214 1.1024 \pm 0.0245	5.0167 \pm 0.0527 1.0578 \pm 0.0432	5.0214 \pm 0.0642 0.9167 \pm 0.0478	5.0512 \pm 0.0805 0.9576 \pm 0.0405	5.0126 \pm 0.6478 0.9432 \pm 0.0489	4.9975 \pm 0.5423 0.9863 \pm 0.0543
$\varepsilon = 2\%$	5.0143 \pm 0.0852 1.0143 \pm 0.0452	5.0278 \pm 0.0452 0.9945 \pm 0.0373	4.9877 \pm 0.0652 1.0345 \pm 0.0878	4.9963 \pm 0.0742 1.1135 \pm 0.0642	5.2342 \pm 0.0879 1.2012 \pm 0.0612	5.3475 \pm 0.8945 1.0286 \pm 0.0602	5.1478 \pm 0.0978 1.0478 \pm 0.0541	5.1478 \pm 0.0978 1.0781 \pm 0.0345	5.1245 \pm 0.1024 0.9977 \pm 0.0536	5.3214 \pm 0.1345 0.9876 \pm 0.0547	5.0245 \pm 0.1178 0.9496 \pm 0.0843
$\varepsilon = 0\%$	5.5679 \pm 0.1352 1.1204 \pm 0.0452	5.3214 \pm 0.1147 1.2354 \pm 0.0987	5.7836 \pm 0.1452 1.2245 \pm 0.0875	5.3245 \pm 0.0698 1.3012 \pm 0.1020	5.4312 \pm 0.1678 1.1546 \pm 0.0834	5.3369 \pm 0.1269 1.1136 \pm 0.8542	5.7548 \pm 0.1689 1.1345 \pm 0.6321	5.8963 \pm 0.1352 1.9989 \pm 0.1040	5.6312 \pm 0.0973 1.1032 \pm 0.0879	5.3475 \pm 0.1042 1.1346 \pm 0.4678	5.4875 \pm 0.1682 1.2320 \pm 0.0843

TABLE 3: Computational time of attribute reduction with using nested reduction or not (mean \pm std. deviation) on *E. coli*.

<i>E. coli</i>	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.2$	$\lambda = 0.3$	$\lambda = 0.4$	$\lambda = 0.5$	$\lambda = 0.6$	$\lambda = 0.7$	$\lambda = 0.8$	$\lambda = 0.9$	$\lambda = 1.0$
$\epsilon = 10\%$	2.3705 \pm 0.0848 0.1637 \pm 0.0159	2.4221 \pm 0.1113 0.1713 \pm 0.0104	2.4765 \pm 0.1118 0.169 \pm 0.0043	2.4578 \pm 0.1368 0.1731 \pm 0.0046	2.4517 \pm 0.1438 0.1724 \pm 0.0048	2.3782 \pm 0.0913 0.1782 \pm 0.0082	2.432 \pm 0.0598 0.1762 \pm 0.0051	3.0164 \pm 0.4828 0.1829 \pm 0.0058	3.0452 \pm 0.2272 0.1814 \pm 0.0035	2.7472 \pm 0.2278 0.1886 \pm 0.0095	1.2564 \pm 0.1585 0.1597 \pm 0.0060
$\epsilon = 8\%$	2.443 \pm 0.0896 0.211 \pm 0.0860	2.5058 \pm 0.0988 0.2167 \pm 0.0975	2.543 \pm 0.0858 0.1927 \pm 0.0841	2.5724 \pm 0.0907 0.228 \pm 0.0984	2.5684 \pm 0.1173 0.2276 \pm 0.1136	2.593 \pm 0.1220 0.299 \pm 0.1589	2.498 \pm 0.1389 0.296 \pm 0.1606	2.4141 \pm 0.0143 0.1948 \pm 0.0140	2.4395 \pm 0.0133 0.1876 \pm 0.0103	2.4392 \pm 0.1348 0.1849 \pm 0.0052	0.9964 \pm 0.0068 0.1591 \pm 0.0087
$\epsilon = 6\%$	2.431 \pm 0.0996 0.1845 \pm 0.0587	2.6975 \pm 0.2672 0.2175 \pm 0.0647	2.7796 \pm 0.2926 0.1659 \pm 0.0028	3.0122 \pm 0.1157 0.199 \pm 0.0063	2.9942 \pm 0.1071 0.1971 \pm 0.0920	3.028 \pm 0.1403 0.2577 \pm 0.1174	3.082 \pm 0.1236 0.3921 \pm 0.1659	3.0489 \pm 0.1561 0.4266 \pm 0.2073	2.8670 \pm 0.0287 0.2089 \pm 0.0046	2.9031 \pm 0.1508 0.2312 \pm 0.1243	1.1553 \pm 0.0127 0.1552 \pm 0.0065
$\epsilon = 4\%$	3.022 \pm 0.1599 0.3286 \pm 0.0875	3.1649 \pm 0.1087 0.3915 \pm 0.1491	3.200 \pm 0.0939 0.330 \pm 0.0653	3.190 \pm 0.1181 0.3973 \pm 0.1498	3.0848 \pm 0.0373 0.1956 \pm 0.0841	3.104 \pm 0.0302 0.233 \pm 0.0985	2.6788 \pm 0.1522 0.2022 \pm 0.0060	3.1241 \pm 0.1087 0.3091 \pm 0.1659	3.0336 \pm 0.1852 0.4717 \pm 0.1923	2.8982 \pm 0.0721 0.1871 \pm 0.0041	1.1558 \pm 0.0135 0.1558 \pm 0.0071
$\epsilon = 2\%$	3.1095 \pm 0.1766 0.2062 \pm 0.0604	3.1861 \pm 0.0943 0.2202 \pm 0.0616	3.2045 \pm 0.0733 0.189 \pm 0.0528	3.249 \pm 0.0563 0.240 \pm 0.0831	3.2665 \pm 0.0861 0.3248 \pm 0.0556	3.291 \pm 0.1091 0.3705 \pm 0.0871	2.9392 \pm 0.2389 0.235 \pm 0.1022	2.7398 \pm 0.1351 0.2411 \pm 0.1057	3.1648 \pm 0.1792 0.3140 \pm 0.1679	3.2586 \pm 0.0496 0.4970 \pm 0.0104	1.1197 \pm 0.0911 0.1592 \pm 0.0094
$\epsilon = 0\%$	3.9261 \pm 0.2688 0.3877 \pm 0.0052	3.2094 \pm 0.0744 0.4030 \pm 0.0052	3.381 \pm 0.0838 0.3799 \pm 0.0075	3.3222 \pm 0.0879 0.401 \pm 0.0058	3.3427 \pm 0.0921 0.4742 \pm 0.0040	3.356 \pm 0.0829 0.4304 \pm 0.0673	3.303 \pm 0.0918 0.395 \pm 0.0688	3.3516 \pm 0.0940 0.5046 \pm 0.0693	3.3662 \pm 0.0775 0.5085 \pm 0.0704	3.492 \pm 0.0773 0.5968 \pm 0.0187	1.1840 \pm 0.0137 0.1773 \pm 0.0088

TABLE 4: Computational time of attribute reduction with using nested reduction or not (mean \pm std. deviation) on Glass.

Glass	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.2$	$\lambda = 0.3$	$\lambda = 0.4$	$\lambda = 0.5$	$\lambda = 0.6$	$\lambda = 0.7$	$\lambda = 0.8$	$\lambda = 0.9$	$\lambda = 1.0$
$\varepsilon = 10\%$	1.5656 \pm 0.0743 0.0657 \pm 0.0065	1.6236 \pm 0.0834 0.0674 \pm 0.0073	1.6282 \pm 0.0905 0.0703 \pm 0.0081	1.6455 \pm 0.0923 0.0749 \pm 0.0068	1.6063 \pm 0.0816 0.0720 \pm 0.0080	1.653 \pm 0.0908 0.0749 \pm 0.0065	1.606 \pm 0.0580 0.0766 \pm 0.0048	1.6047 \pm 0.0128 0.0781 \pm 0.0003	1.6440 \pm 0.0283 0.0812 \pm 0.0067	1.5314 \pm 0.0824 0.0796 \pm 0.0051	1.2579 \pm 0.1693 0.1938 \pm 0.0974
$\varepsilon = 8\%$	1.601 \pm 0.0385 0.1125 \pm 0.0735	1.6610 \pm 0.0576 0.1219 \pm 0.0781	1.6766 \pm 0.0579 0.118 \pm 0.0800	1.6907 \pm 0.0580 0.1250 \pm 0.0754	1.6845 \pm 0.0739 0.1156 \pm 0.0750	1.712 \pm 0.0490 0.151 \pm 0.0909	1.7376 \pm 0.0593 0.125 \pm 0.0832	1.6360 \pm 0.0769 0.0954 \pm 0.0543	1.6472 \pm 0.0778 0.1313 \pm 0.0895	1.6390 \pm 0.0922 0.1719 \pm 0.0990	1.2595 \pm 0.1678 0.0719 \pm 0.0079
$\varepsilon = 6\%$	1.613 \pm 0.0066 0.081 \pm 0.0488	1.6813 \pm 0.0168 0.0922 \pm 0.0454	1.6923 \pm 0.0128 0.0860 \pm 0.0525	1.7032 \pm 0.0129 0.084 \pm 0.0480	1.7189 \pm 0.0105 0.0906 \pm 0.0508	1.7423 \pm 0.0114 0.1109 \pm 0.0692	1.7501 \pm 0.0107 0.1095 \pm 0.0701	1.7751 \pm 0.0184 0.2142 \pm 0.0717	1.7782 \pm 0.0454 0.2032 \pm 0.0831	1.7112 \pm 0.0421 0.1610 \pm 0.0977	1.2484 \pm 0.1653 0.0703 \pm 0.0081
$\varepsilon = 4\%$	1.601 \pm 0.0082 0.064 \pm 0.0049	1.6704 \pm 0.0115 0.0704 \pm 0.0080	1.684 \pm 0.0122 0.071 \pm 0.0112	1.7110 \pm 0.0372 0.0720 \pm 0.0080	1.7125 \pm 0.0082 0.0844 \pm 0.0368	1.734 \pm 0.0103 0.0782 \pm 0.0004	1.7486 \pm 0.0168 0.0782 \pm 0.0004	1.7657 \pm 0.0146 0.0766 \pm 0.0051	1.7689 \pm 0.0429 0.0830 \pm 0.0076	1.7188 \pm 0.0655 0.0860 \pm 0.0080	1.2454 \pm 0.1638 0.0719 \pm 0.0081
$\varepsilon = 2\%$	1.6767 \pm 0.0590 0.128 \pm 0.0565	1.7048 \pm 0.0497 0.0904 \pm 0.0516	1.7032 \pm 0.0353 0.084 \pm 0.0369	1.736 \pm 0.0455 0.084 \pm 0.0369	1.7438 \pm 0.0322 0.0845 \pm 0.0370	1.7408 \pm 0.0197 0.0782 \pm 0.0004	1.7610 \pm 0.0180 0.0765 \pm 0.0047	1.7783 \pm 0.0161 0.0780 \pm 0.0000	1.7954 \pm 0.0258 0.0812 \pm 0.0067	1.7235 \pm 0.0714 0.0845 \pm 0.0080	1.2469 \pm 0.1647 0.0686 \pm 0.0083
$\varepsilon = 0\%$	1.797 \pm 0.0146 0.1780 \pm 0.0521	1.8814 \pm 0.0186 0.2483 \pm 0.0600	1.886 \pm 0.0149 0.253 \pm 0.0341	1.9142 \pm 0.0200 0.2578 \pm 0.0360	1.9281 \pm 0.0170 0.2469 \pm 0.0479	1.946 \pm 0.0153 0.2797 \pm 0.0052	1.9672 \pm 0.0172 0.281 \pm 0.0004	1.9892 \pm 0.0129 0.2874 \pm 0.0083	2.0171 \pm 0.0281 0.2985 \pm 0.0114	2.1236 \pm 0.0326 0.3328 \pm 0.0616	1.3546 \pm 0.1654 0.2718 \pm 0.0082

TABLE 5: Computational time of attribute reduction with using nested reduction or not (mean \pm std. deviation) on Iris.

Iris	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.2$	$\lambda = 0.3$	$\lambda = 0.4$	$\lambda = 0.5$	$\lambda = 0.6$	$\lambda = 0.7$	$\lambda = 0.8$	$\lambda = 0.9$	$\lambda = 1.0$
$\varepsilon = 10\%$	0.1266 \pm 0.0051	0.1282 \pm 0.0145	0.100 \pm 0.0080	0.095 \pm 0.0186	0.0516 \pm 0.0076	0.112 \pm 0.0422	0.1423 \pm 0.0092	0.1297 \pm 0.0197	0.1016 \pm 0.0082	0.1219 \pm 0.0220	0.0969 \pm 0.0064
	0.021 \pm 0.0084	0.0265 \pm 0.0077	0.0234 \pm 0.0082	0.0250 \pm 0.0082	0.0328 \pm 0.0216	0.022 \pm 0.0431	0.051 \pm 0.0483	0.0266 \pm 0.0076	0.0250 \pm 0.0080	0.0268 \pm 0.0178	0.0375 \pm 0.0198
$\varepsilon = 8\%$	0.1266 \pm 0.0051	0.1344 \pm 0.0081	0.0968 \pm 0.0067	0.092 \pm 0.0171	0.0532 \pm 0.0082	0.139 \pm 0.0050	0.134 \pm 0.0081	0.1359 \pm 0.0147	0.1063 \pm 0.0163	0.1250 \pm 0.0219	0.0969 \pm 0.0067
	0.023 \pm 0.0083	0.0266 \pm 0.0076	0.0250 \pm 0.0082	0.023 \pm 0.0080	0.0391 \pm 0.0211	0.049 \pm 0.0380	0.0251 \pm 0.0078	0.0266 \pm 0.0073	0.0251 \pm 0.0081	0.0281 \pm 0.0145	0.0250 \pm 0.0082
$\varepsilon = 6\%$	0.1250 \pm 0.0000	0.1360 \pm 0.0076	0.123 \pm 0.0214	0.101 \pm 0.0149	0.0579 \pm 0.0245	0.139 \pm 0.0050	0.139 \pm 0.0046	0.1360 \pm 0.0076	0.1156 \pm 0.0210	0.1328 \pm 0.0154	0.0938 \pm 0.0004
	0.018 \pm 0.0064	0.0250 \pm 0.0082	0.023 \pm 0.0081	0.0235 \pm 0.0079	0.0438 \pm 0.0293	0.0220 \pm 0.0082	0.0266 \pm 0.0076	0.0266 \pm 0.0076	0.0266 \pm 0.0076	0.0360 \pm 0.0199	0.0234 \pm 0.0084
$\varepsilon = 4\%$	0.1407 \pm 0.0073	0.1563 \pm 0.0128	0.148 \pm 0.0324	0.1078 \pm 0.0200	0.1406 \pm 0.0436	0.139 \pm 0.0049	0.1375 \pm 0.0066	0.1375 \pm 0.0066	0.1360 \pm 0.0209	0.1391 \pm 0.0088	0.0938 \pm 0.0004
	0.0275 \pm 0.0110	0.0500 \pm 0.0100	0.0278 \pm 0.0193	0.0250 \pm 0.0077	0.0531 \pm 0.0245	0.0297 \pm 0.0088	0.0265 \pm 0.0075	0.0281 \pm 0.0067	0.0281 \pm 0.0099	0.0328 \pm 0.0227	0.0234 \pm 0.0087
$\varepsilon = 2\%$	0.1454 \pm 0.0105	0.1595 \pm 0.0066	0.1626 \pm 0.0079	0.1173 \pm 0.0347	0.1422 \pm 0.0467	0.1562 \pm 0.0147	0.1376 \pm 0.0144	0.1375 \pm 0.0144	0.1562 \pm 0.0196	0.1390 \pm 0.0088	0.0954 \pm 0.0051
	0.0266 \pm 0.0107	0.0297 \pm 0.0139	0.0391 \pm 0.0307	0.036 \pm 0.0313	0.0312 \pm 0.0073	0.0483 \pm 0.0137	0.0266 \pm 0.0105	0.0281 \pm 0.0145	0.0316 \pm 0.0196	0.0250 \pm 0.0080	0.0234 \pm 0.0080
$\varepsilon = 0\%$	0.1469 \pm 0.0107	0.1625 \pm 0.0080	0.1725 \pm 0.0138	0.1594 \pm 0.0067	0.1641 \pm 0.0081	0.164 \pm 0.0083	0.1610 \pm 0.0076	0.1610 \pm 0.0071	0.1626 \pm 0.0081	0.1476 \pm 0.0099	0.0989 \pm 0.0064
	0.035 \pm 0.0077	0.0381 \pm 0.0069	0.0351 \pm 0.0078	0.0766 \pm 0.0289	0.0581 \pm 0.0067	0.051 \pm 0.0103	0.0556 \pm 0.0133	0.0531 \pm 0.0153	0.0543 \pm 0.0232	0.0350 \pm 0.0082	0.0465 \pm 0.0079

TABLE 6: Computational time of attribute reduction with using nested reduction or not (mean \pm std. deviation) on Seeds.

Seeds	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.2$	$\lambda = 0.3$	$\lambda = 0.4$	$\lambda = 0.5$	$\lambda = 0.6$	$\lambda = 0.7$	$\lambda = 0.8$	$\lambda = 0.9$	$\lambda = 1.0$
$\varepsilon = 10\%$	0.4375 \pm 0.0103 0.039 \pm 0.0081	0.5250 \pm 0.0397 0.0469 \pm 0.0003	0.554 \pm 0.0325 0.046 \pm 0.0004	0.534 \pm 0.0434 0.0470 \pm 0.0000	0.5547 \pm 0.0404 0.0485 \pm 0.0047	0.5375 \pm 0.0108 0.048 \pm 0.0048	0.5720 \pm 0.0257 0.0500 \pm 0.0066	0.5594 \pm 0.0217 0.0517 \pm 0.0076	0.5766 \pm 0.0669 0.0562 \pm 0.0079	0.4609 \pm 0.0221 0.0530 \pm 0.0080	0.6030 \pm 0.0290 0.0475 \pm 0.0249
$\varepsilon = 8\%$	0.4328 \pm 0.0075 0.039 \pm 0.0084	0.4938 \pm 0.0080 0.0468 \pm 0.0004	0.5032 \pm 0.0067 0.046 \pm 0.0004	0.510 \pm 0.0105 0.0452 \pm 0.0050	0.5297 \pm 0.0048 0.0484 \pm 0.0051	0.532 \pm 0.0050 0.0516 \pm 0.0074	0.5531 \pm 0.0081 0.0531 \pm 0.0079	0.5422 \pm 0.0073 0.0499 \pm 0.0067	0.5594 \pm 0.0098 0.0531 \pm 0.0081	0.4266 \pm 0.0076 0.0547 \pm 0.0081	0.5938 \pm 0.0304 0.0470 \pm 0.0231
$\varepsilon = 6\%$	0.5063 \pm 0.0249 0.040 \pm 0.0300	0.5782 \pm 0.0282 0.0565 \pm 0.0342	0.5923 \pm 0.0333 0.035 \pm 0.0324	0.6047 \pm 0.0376 0.0377 \pm 0.0328	0.6171 \pm 0.0364 0.0380 \pm 0.0393	0.623 \pm 0.0331 0.043 \pm 0.0344	0.5470 \pm 0.0000 0.0515 \pm 0.0077	0.653 \pm 0.0480 0.0516 \pm 0.0375	0.5547 \pm 0.0081 0.0532 \pm 0.0080	0.4234 \pm 0.0048 0.0500 \pm 0.0066	0.5922 \pm 0.0290 0.0422 \pm 0.0077
$\varepsilon = 4\%$	0.5188 \pm 0.0230 0.0515 \pm 0.0277	0.5969 \pm 0.0274 0.0657 \pm 0.0404	0.6187 \pm 0.0288 0.0734 \pm 0.0398	0.6204 \pm 0.0074 0.0579 \pm 0.0348	0.6297 \pm 0.0076 0.0719 \pm 0.0450	0.6437 \pm 0.0065 0.060 \pm 0.0340	0.6579 \pm 0.0050 0.1656 \pm 0.0076	0.6469 \pm 0.0078 0.059 \pm 0.0343	0.5517 \pm 0.0105 0.0515 \pm 0.0075	0.5001 \pm 0.0641 0.0485 \pm 0.0047	0.5906 \pm 0.0321 0.0421 \pm 0.0077
$\varepsilon = 2\%$	0.5813 \pm 0.0273 0.089 \pm 0.0232	0.6843 \pm 0.0529 0.1093 \pm 0.0345	0.7015 \pm 0.0379 0.1047 \pm 0.0370	0.6859 \pm 0.0278 0.1203 \pm 0.0265	0.7078 \pm 0.0077 0.1249 \pm 0.0284	0.7219 \pm 0.0067 0.134 \pm 0.0081	0.7392 \pm 0.0301 0.1376 \pm 0.0276	0.735 \pm 0.0050 0.142 \pm 0.0049	0.6672 \pm 0.0077 0.1703 \pm 0.0089	0.5516 \pm 0.0076 0.0765 \pm 0.0508	0.6172 \pm 0.0211 0.0640 \pm 0.0228
$\varepsilon = 0\%$	0.606 \pm 0.0099 0.0892 \pm 0.0210	0.7142 \pm 0.0147 0.1062 \pm 0.0242	0.7375 \pm 0.0066 0.1094 \pm 0.0287	0.753 \pm 0.0065 0.1287 \pm 0.0245	0.7672 \pm 0.0052 0.1341 \pm 0.0264	0.7859 \pm 0.0077 0.149 \pm 0.0005	0.8127 \pm 0.0105 0.1450 \pm 0.0285	0.7985 \pm 0.0047 0.158 \pm 0.0080	0.8189 \pm 0.0078 0.2094 \pm 0.0083	0.8219 \pm 0.0083 0.3186 \pm 0.0083	0.6423 \pm 0.0048 0.0709 \pm 0.0260

TABLE 7: Computational time of attribute reduction with using nested reduction or not (mean \pm std. deviation) on Wine.

Iris	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.2$	$\lambda = 0.3$	$\lambda = 0.4$	$\lambda = 0.5$	$\lambda = 0.6$	$\lambda = 0.7$	$\lambda = 0.8$	$\lambda = 0.9$	$\lambda = 1.0$
$\varepsilon = 10\%$	1.4704 \pm 0.0677 0.0376 \pm 0.0077	1.5689 \pm 0.0698 0.0422 \pm 0.0075	1.6157 \pm 0.0716 0.0407 \pm 0.0079	1.6173 \pm 0.0649 0.045 \pm 0.0050	1.6406 \pm 0.0208 0.0470 \pm 0.0000	1.5126 \pm 0.0987 0.050 \pm 0.0066	1.383 \pm 0.1115 0.048 \pm 0.0048	1.3283 \pm 0.0966 0.0501 \pm 0.0068	1.422 \pm 0.1012 0.0515 \pm 0.0077	0.8843 \pm 0.1265 0.0484 \pm 0.0051	1.0656 \pm 0.0815 0.0515 \pm 0.1160
$\varepsilon = 8\%$	1.484 \pm 0.0163 0.0876 \pm 0.0662	1.6109 \pm 0.0201 0.0984 \pm 0.0773	1.6783 \pm 0.0257 0.1094 \pm 0.0776	1.6907 \pm 0.0579 0.0781 \pm 0.0622	1.6439 \pm 0.0708 0.0781 \pm 0.0658	1.604 \pm 0.0889 0.0485 \pm 0.0047	1.459 \pm 0.0996 0.048 \pm 0.0048	1.3548 \pm 0.0129 0.0501 \pm 0.0065	1.1547 \pm 0.0716 0.0485 \pm 0.0051	0.9766 \pm 0.0108 0.0515 \pm 0.0075	1.0579 \pm 0.0850 0.0422 \pm 0.0073
$\varepsilon = 6\%$	1.5720 \pm 0.0322 0.123 \pm 0.0452	1.6969 \pm 0.0496 0.1390 \pm 0.0534	1.775 \pm 0.0539 0.1407 \pm 0.0539	1.779 \pm 0.0651 0.1437 \pm 0.0675	1.7766 \pm 0.0600 0.1547 \pm 0.0745	1.653 \pm 0.0896 0.0657 \pm 0.0541	1.512 \pm 0.1045 0.050 \pm 0.0069	1.3751 \pm 0.0784 0.0500 \pm 0.0066	1.1860 \pm 0.0212 0.0500 \pm 0.0066	0.9750 \pm 0.0151 0.0515 \pm 0.0072	1.1048 \pm 0.0899 0.0437 \pm 0.0067
$\varepsilon = 4\%$	1.717 \pm 0.1201 0.112 \pm 0.0353	1.7799 \pm 0.0497 0.1265 \pm 0.0431	1.8641 \pm 0.0537 0.1170 \pm 0.0532	1.8907 \pm 0.0603 0.1328 \pm 0.0607	1.9204 \pm 0.0263 0.1547 \pm 0.0572	1.8549 \pm 0.0970 0.209 \pm 0.0083	1.841 \pm 0.0943 0.2391 \pm 0.0075	1.7579 \pm 0.3240 0.1141 \pm 0.1047	1.4310 \pm 0.1112 0.0750 \pm 0.0783	1.1108 \pm 0.0629 0.0516 \pm 0.0079	1.2240 \pm 0.1055 0.0437 \pm 0.0067
$\varepsilon = 2\%$	1.9824 \pm 0.0783 0.132 \pm 0.0469	2.1187 \pm 0.0896 0.165 \pm 0.0496	2.132 \pm 0.0535 0.168 \pm 0.0429	2.1876 \pm 0.0853 0.168 \pm 0.0429	2.2345 \pm 0.0943 0.1783 \pm 0.0390	2.232 \pm 0.1300 0.2078 \pm 0.0473	1.9678 \pm 0.1700 0.184 \pm 0.0693	1.8470 \pm 0.2499 0.2126 \pm 0.0875	1.6112 \pm 0.2811 0.0937 \pm 0.0949	1.1905 \pm 0.1081 0.0969 \pm 0.1055	1.1880 \pm 0.1564 0.0437 \pm 0.0067
$\varepsilon = 0\%$	2.116 \pm 0.0541 0.1875 \pm 0.0198	2.9588 \pm 0.0430 0.2078 \pm 0.0299	2.232 \pm 0.0179 0.185 \pm 0.0354	2.294 \pm 0.0222 0.189 \pm 0.0421	2.2506 \pm 0.0206 0.1783 \pm 0.0482	2.3263 \pm 0.0271 0.2750 \pm 0.0539	2.293 \pm 0.0456 0.503 \pm 0.0851	2.2867 \pm 0.0493 0.6843 \pm 0.1012	1.7365 \pm 0.1552 0.3359 \pm 0.1029	2.4604 \pm 0.1158 0.2656 \pm 0.0772	1.4571 \pm 0.0784 0.0821 \pm 0.0077

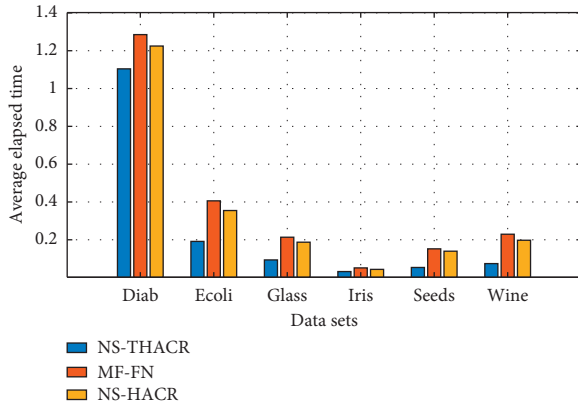


FIGURE 1: Elapsed time of three different algorithms.

xx axis denotes different data sets and yy is the average elapsed time of three different algorithms. It is difficult to note from Figure 1 that the proposed method NS-THACR obtained the minimum elapsed time. With the above discussions, we can conclude that the proposed NS-THACR method is an effective and efficient attribute reduction method.

5. Conclusions

In this paper, we have discussed some weaknesses in attribute reduction based on the traditional fuzzy rough set, which is often constructed by the operator without adjustability, and thereby a new adjustable fuzzy rough set has been presented. In our approach, a parameterized operator has been applied to develop strong and weak fuzzy relations, and such two special relations can offer fuzzy rough set model adjustability. Furthermore, inspired by the inner relationship between parameter and reduct, we have also employed a nested mechanism to accelerate the searching process of parameterized reduct.

The following topics deserve our further investigations:

(1) some evolutionary algorithms [2] such as particle swarm optimization algorithm and ant colony algorithm can be applied to find the optimal parameter used in our proposed model; (2) some accelerated searching strategy [12] such as sampling technique and attribute group technique can be used to further reduce the elapsed time over one single parameter.

Data Availability

To evaluate the performance of our nested strategy-based heuristic attribute reduction approach, 6 real-world UCI data sets have been employed in this paper. Table 1 summarizes some details of these experimental data sets. All the experiments are run on a workstation equipped with a 3.10 GHz processor and a 8.00 G memory. The programming language is Matlab R2014b.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61503229, the Natural Science Foundation of Shanxi Province of China under Grant No. 201901D111033, and the Key Research and Development Project of Shanxi Province under Grant No. 201803D31055.

References

- [1] Z. a. Pawlak, "Rough sets," *International Journal of Computer & Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.
- [2] H. Chen, T. Li, X. Fan, and C. Luo, "Feature selection for imbalanced data based on neighborhood rough sets," *Information Sciences*, vol. 483, pp. 1–20, 2019.
- [3] C. Wang, Y. Huang, W. Ding, and Z. Cao, "Attribute reduction with fuzzy rough self-information measures," *Information Sciences*, vol. 549, pp. 68–86, 2021.
- [4] H. Ju, H. Li, X. Yang, X. Zhou, and B. Huang, "Cost-sensitive rough set: a multi-granulation approach," *Knowledge-Based Systems*, vol. 123, pp. 137–153, 2017.
- [5] H. Ju, X. Yang, H. Yu, T. Li, D.-J. Yu, and J. Yang, "Cost-sensitive rough set approach," *Information Sciences*, vol. 355–356, pp. 282–298, 2016.
- [6] K. Liu, X. Yang, H. Fujita, D. Liu, X. Yang, and Y. Qian, "An efficient selector for multi-granularity attribute reduction," *Information Sciences*, vol. 505, pp. 457–472, 2019.
- [7] K. Liu, X. Yang, H. Yu, J. Mi, P. Wang, and X. Chen, "Rough set based semi-supervised feature selection via ensemble selector," *Knowledge-Based Systems*, vol. 165, pp. 282–296, 2019.
- [8] W. Shu and H. Shen, "Incremental feature selection based on rough set in dynamic incomplete data," *Pattern Recognition*, vol. 47, no. 12, pp. 3890–3906, 2014.
- [9] S. Xu, X. Yang, H. Yu, D.-J. Yu, J. Yang, and E. C. C. Tsang, "Multi-label learning with label-specific feature reduction," *Knowledge-Based Systems*, vol. 104, pp. 52–61, 2016.
- [10] J. Dai and J. Chen, "Feature selection via normative fuzzy information weight with application into tumor classification," *Applied Soft Computing*, vol. 92, Article ID 106299, 2020.
- [11] C. Wang, Y. Huang, M. Shao, Q. Hu, and D. Chen, "Feature selection based on neighborhood self-information," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 4031–4042, 2020.
- [12] Y. Qian, Q. Wang, H. Cheng, J. Liang, and C. Dang, "Fuzzy-rough feature selection accelerator," *Fuzzy Sets and Systems*, vol. 258, pp. 61–78, 2015.
- [13] Y. Yang, D. Chen, H. Wang, E. C. C. Tsang, and D. Zhang, "Fuzzy rough set based incremental attribute reduction from dynamic data with sample arriving," *Fuzzy Sets and Systems*, vol. 312, pp. 66–86, 2017.
- [14] A. Zeng, T. Li, D. Liu, J. Zhang, and H. Chen, "A fuzzy rough set approach for incremental feature selection on hybrid information systems," *Fuzzy Sets and Systems*, vol. 258, pp. 39–60, 2015.
- [15] X. Zhang, C. Mei, D. Chen, Y. Yang, and J. Li, "Active incremental feature selection using a fuzzy-rough-set-based information entropy," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 5, pp. 901–915, 2020.
- [16] J. Chen, J. Mi, and Y. Lin, "A graph approach for fuzzy-rough feature selection," *Fuzzy Sets and Systems*, vol. 391, pp. 96–116, 2020.

- [17] L. W. Fung and K. S. Fu, "An axiomatic approach to rational decision making in a fuzzy environment††this work was supported by the national science foundation grant GK-36721," in *Proceedings of the Fuzzy Sets and Their Applications to Cognitive and Decision Processes*, pp. 227–256, Berkeley, CA, USA, July 1975.
- [18] R. Jensen and N. Mac Parthaláin, "Towards scalable fuzzy-rough feature selection," *Information Sciences*, vol. 323, pp. 1–15, 2015.
- [19] Y. Li, Y. Lin, J. Liu, W. Weng, Z. Shi, and S. Wu, "Feature selection for multi-label learning based on kernelized fuzzy rough sets," *Neurocomputing*, vol. 318, pp. 271–286, 2018.
- [20] X. Zhang, C. Mei, D. Chen, and Y. Yang, "A fuzzy rough set-based feature selection method using representative instances," *Knowledge-Based Systems*, vol. 151, pp. 216–229, 2018.
- [21] Q. Hu, L. Zhang, D. Chen, W. Pedrycz, and D. Yu, "Gaussian kernel based fuzzy rough sets: model, uncertainty measures and applications," *International Journal of Approximate Reasoning*, vol. 51, no. 4, pp. 453–471, 2010.
- [22] Q. Hu, L. Zhang, S. An, D. Zhang, and D. Yu, "On robust fuzzy rough set models," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 4, pp. 636–651, 2012.
- [23] J. Ye, J. Zhan, W. Ding, and H. Fujita, "A novel fuzzy rough set model with fuzzy neighborhood operators," *Information Sciences*, vol. 544, pp. 266–297, 2021.
- [24] D. Dubois and H. Prade, "Rough fuzzy sets and fuzzy rough sets*," *International Journal of General Systems*, vol. 17, no. 2–3, pp. 191–209, 1990.
- [25] C. Wang, Y. Wang, M. Shao, Y. Qian, and D. Chen, "Fuzzy rough attribute reduction for categorical data," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 5, pp. 818–830, 2020.
- [26] N. Parthaláin, Q. Shen, and R. Jensen, "A distance measure approach to exploring the rough set boundary region for attribute reduction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 3, pp. 305–317, 2010.
- [27] D. Slezak, "Rough sets and functional dependencies in data: foundations of association reducts," *Transactions on Computational Science*, vol. 5, pp. 182–205, 2009.
- [28] D. Slezak, "Approximate entropy reducts," *Fundamenta Informaticae*, vol. 53, pp. 365–390, 2002.
- [29] F. Wang, J. Liang, and Y. Qian, "Attribute reduction: a dimension incremental strategy," *Knowledge-Based Systems*, vol. 39, pp. 95–108, 2013.
- [30] G. Wang, X. A. Ma, and H. Yu, "Monotonic uncertainty measures for attribute reduction in probabilistic rough set model," *International Journal of Approximate Reasoning*, vol. 59, pp. 41–67, 2015.
- [31] Q. H. Qinghua Hu, W. Pedrycz, D. Yu, and J. Jun Lang, "Selecting discrete and continuous features based on neighborhood decision error minimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 1, pp. 137–150, 2010.
- [32] X. Yang, Y. Qi, X. Song, and J. Yang, "Test cost sensitive multigranulation rough set: model and minimal cost selection," *Information Sciences*, vol. 250, pp. 184–199, 2013.
- [33] X. Yang, Y. Qi, H. Yu, X. Song, and J. Yang, "Updating multigranulation rough approximations with increasing of granular structures," *Knowledge-Based Systems*, vol. 64, pp. 59–69, 2014.
- [34] K. Zheng, J. Hu, Z. Zhan, J. Ma, and J. Qi, "An enhancement for heuristic attribute reduction algorithm in rough set," *Expert Systems with Applications*, vol. 41, no. 15, pp. 6748–6754, 2014.
- [35] X. Yang, S. Xu, H. Dou, X. Song, H. Yu, and J. Yang, "Multigranulation rough set: a multiset based strategy," *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 277–292, 2017.
- [36] S. Zhao, X. Wang, D. Chen, and E. C. C. Tsang, "Nested structure in parameterized rough reduction," *Information Sciences*, vol. 248, pp. 130–150, 2013.
- [37] K. Liu, X. Yang, H. Yu et al., "Supervised information granulation strategy for attribute reduction," *International Journal of Machine Learning and Cybernetics*, vol. 11, pp. 2149–2163, 2020.