Information Sciences 397–398 (2017) 15–36

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Quick attribute reduction with generalized indiscernibility models

Fan Jing[a,b], Jiang Yunliang[c], Liu Yong[a,b,*]

[a] Institute of Cyber Systems and Control, Zhejiang University, 310027 Hangzhou, China
[b] State Key Laboratory of Industrial Control and Technology, Zhejiang University, 310027 Hangzhou, China
[c] Huzhou University, 313000 Huzhou, China

ARTICLE INFO

Article history:
Received 27 May 2016
Revised 26 December 2016
Accepted 14 February 2017
Available online 20 February 2017

Keywords:
Generalized indiscernibility relation
Attribute reduction
Granular structure
Acceleration policy

ABSTRACT

The efficiency of attribute reduction is one of the important challenges being faced in the field of Big Data processing. Although many quick attribute reduction algorithms have been proposed, they are tightly coupled with their corresponding indiscernibility relations, and it is difficult to extend specific acceleration policies to other reduction models. In this paper, we propose a generalized indiscernibility reduction model(GIRM) and a concept of the granular structure in GIRM, which is a quantitative measurement induced from multiple indiscernibility relations and which can be used to represent the computation cost of varied models. Then, we prove that our GIRM is compatible with three typical reduction models. Based on the proposed GIRM, we present a generalized attribute reduction algorithm and a generalized positive region computing algorithm. We perform a quantitative analysis of the computation complexities of two algorithms using the granular structure. For the generalized attribute reduction, we present systematic acceleration policies that can reduce the computational domain and optimize the computation of the positive region. Based on the granular structure, we propose acceleration policies for the computation of the generalized positive region, and we also propose fast positive region computation approaches for three typical reduction models. Experimental results for various datasets prove the efficiency of our acceleration policies in those three typical reduction models.

© 2017 Elsevier Inc. All rights reserved.

1. Introduction

The processing of Big Data always suffers from the high dimensional attributes and large number of the objects. Thus, the reduction [17,20,22,43] has become a very important preprocessing method employed to remove the redundancy on those high dimensional attributes in Big Data applications. As reduction can decrease the computation complexity, improve the data quality, and promote the accuracy of the classifiers, it is widely used in data mining [31,45], machine learning [9,55], and pattern recognition [14,43]. Form the perspective of the indiscernibility, reduction is an attribute subset that has the same distinguishing capability compared with the original attribute set. According to different definitions of the distinguishing relations (also denoted as indiscernibility relations), various reduction models have been proposed, such as the classical rough set reduction model [43,61], neighborhood rough set reduction model [13,17], and fuzzy rough set reduction model [1,12,32].

* Corresponding author.
E-mail address: cckaffe@hotmail.com (L. Yong).

http://dx.doi.org/10.1016/j.ins.2017.02.032
0020-0255/© 2017 Elsevier Inc. All rights reserved.

Although reduction can greatly improve the performance of Big Data applications, the large dimension of the attributes and the large number of objects remain a challenge that affects the efficiency of current reduction methods. Thus, fast reduction algorithms are expected. With respect to fast reduction algorithms, they closely rely on their corresponding indiscernibility relations, such as the equivalence relation [34], fuzzy relation [51], tolerance relation [33], similarity relation [42], and neighborhood relation [17]. Therefore, the fast algorithm may only work on a specific reduction model that is determined by its indiscernibility relation. Thus, it is difficult to evaluate the overall computation cost in terms of the reduction algorithms under varied indiscernibility relations without a unified reduction model established based on the concept of generalized indiscernibility. Furthermore, for the acceleration polices [10,23,54,61] employed in different fast reduction algorithms with respect to their corresponding indiscernibility relations, it is also difficult for them to be extended into other reduction models without the need for a generalized model.

In this paper, we present a generalized indiscernibility reduction model(GIRM) that addresses the above two difficulties. We also prove that our GIRM is compatible with three typical reduction models, i.e., the equivalent relation, neighborhood relation, and fuzzy relation based reduction models. In our GIRM, we present a concept of the granular structure, which is a quantitative measurement induced from different indiscernibility relations. The granular structure can be used to represent the computation cost of the generalized reduction model under all the indiscernibility relations. Based on GIRM, we present a generalized attribute reduction algorithm as well as a generalized positive region computing algorithm. We quantitatively analyze the generalized computation complexities of those two algorithms using the granular structure presented in GIRM. We then present our acceleration policies for those two generalized algorithms. Specifically, we also propose fast approaches for three typical reduction models, i.e., the classical rough set model, neighborhood rough set model, and fuzzy rough set model, and we perform the corresponding experiments to show that our fast approaches for the three typical reduction models are superior to state-of-the-art methods.

2. Related works

Generally, there are two kinds of attribute reduction approaches, i.e., indiscernibility matrix based methods [6,36,39,40,44,59] and attribute combination(or search policy) based methods [13,20,29,30,61]. Although the former can be used to obtain all of the reductions, the large temporal and spatial complexity of these methods limits their implementation in Big Data applications. The latter normally includes two key issues, the metric function and the search policy. In combination based methods, the metric function is used to evaluate whether the input attribute subset is a reduction, while the search policy is used to generate the candidate attribute subset that will be evaluated by the metric function. There are a variety of search policies, including random [26], enumerating [35], forward greedy [19], and backward greedy [56]. Obviously, the random and enumerating search policies are not suitable for Big Data owing to the high computation cost required to find the attribute reduction. Then, greedy based search policies are dominant when dealing with Big Data. Of the two greedy search policies, the forward greedy search policy is preferred owing to its simplicity and superior efficiency. There are also various metric functions, such as positive region (dependency) [4], inconsistency [29], entropy [47], and combination entropy [24]. Of all those metric functions, the positive region is directly induced from the concept of indiscernibility, and it is therefore widely used in reduction algorithms with respect to various indiscernibility relations.

In order to better apply attribute reduction in practical Big Data applications, many fast attribute reduction algorithms [13,30,38,54] have been proposed. Usually, there are two categories of approaches to reduce the temporal complexity of the reduction algorithms, one is to reduce the computational domain based on the monotonous theory [13,38], while the other is to optimize the computation of the positive region [30,54] that is closely related to the specific rough set model. From the perspective of the evaluation, the first category is based on the property of monotonicity in that the metric function monotonically increases with attributes. This means that the addition of a new attribute to the candidate attribute subset will not decrease the metric function. The property of monotonicity induces if an object x belongs to the positive region with respect to an attribute set A, then, for $B \supseteq A$, x belongs to the positive region with respect to B. Following this idea, Hu et al. [13] proposed the fast forward heterogeneous attribute reduction algorithm, which is based on the neighborhood rough set model and the fast forward discrete attribute reduction algorithm, which is based on the classical rough set model. Similarly, Qian et al. [37,38] proposed the rank preservation principle to design attribute reduction accelerators based on the classical rough set model and fuzzy rough set model. The principle is that the rank of attributes in the process of attribute reduction remains unchanged after reducing the lower approximation, which can essentially ensure the monotonicity of the metric function. From the perspective of the granulation order, the property of monotonicity can be viewed as the further induction of the rank preservation principle. In fact, the first category of the acceleration approaches can be more common in previous methods as most of the rough set models can satisfy the property of monotonicity. For the second category, which optimizes the computation of the positive region, they closely depend on the specific rough set model. By employing the sort methods, Liu et al. [27] introduced improved algorithms for the computation of the positive region based on the classical rough set model, and Hu et al. [13] proposed an algorithm that is based on the neighborhood rough set model. In the classical rough set model, the equivalence classes are the partitions of the universe, and Liu et al. [30] proposed a quick algorithm based on the hash table that represents the partition, and which can reduce the computation of positive region. In neighborhood rough model, the objects' neighborhoods form the cover instead of the partition of the universe. Then, Liu et al. [54] presented the notion of the hash bucket, dividing objects into a series of sequenced buckets, this enables us to find each object's neighborhood in its own bucket and in its adjacent buckets rather than in the whole universe. In

the fuzzy rough model, Bhatt and Gopal [1] used the property of fuzzy approximation to simplify the computation of the degree of objects belonging to the positive region. Although different methods are implemented in the second category, we find that the commonality of these algorithms is that they actually optimize the computation of the granular structure, which is presented in this paper, and can unify the different rough set models. For static datasets, besides the above two categories of acceleration approaches, researchers have in recent years strived to design attribute reduction methods using parallel computational models. Zhang et al. [58] proposed a parallel method for computing rough set approximations based on the MapReduce technique. For dynamic datasets, the incremental approaches focus mainly on two perspectives, i.e., the objects' variation and the attributes' variation. For example, Wang et al. [46] proposed an incremental reduction algorithm based on information entropy with dynamically increasing attributes. Chen et al. [2] proposed an incremental method for updating the rough set approximations when objects in the universe evolve over time. However, these methods are only limited in rough set models based on the classical equivalence relation, and they lack a unified reduction model and unified acceleration polices that would enable them to be extended to other rough set models.

3. Generalized reduction model

In this section, we present the definition of our generalized model with indiscernibility. We also discuss the relations between our generalized model and three typical rough set models(i.e., the classical rough set model, neighborhood rough set model, and fuzzy rough set model).

3.1. Generalized indiscernibility reduction model

First, we define the decision system used in our generalized indiscernibility reduction model as follows.

A decision system DS can be denoted by $\langle U, A\rangle$, where $U = \{x_1, x_2, \ldots, x_n\}$ is a nonempty and finite set of objects, called the universe, $A = C \cup D$ is a finite set of attributes to characterize the objects, $C = \{a_1, a_2, \ldots, a_m\}$ is the condition attribute set, and D is the decision attribute set.

Indiscernibility reflects the distinguishing possibility among objects with the available information, which is always represented by the relation R defined on U. In the classical rough set, R is simplified to an equivalence relation, which needs to satisfy all of the reflexive, symmetric, and transitive properties. However, the equivalence relation may be too strict to lead to low effects in real cases. To address this problem in practical applications, many extensions [3,7,25,41,42,52,53] of the classical equivalence relation were developed. For example, Hu et al. [13] extended the equivalence relation to the neighborhood relation, which can achieve much better performance in feature selection. Among the classical equivalence relation and its extensions, all of them have the common essential–indiscernibility, which is the degree of the distinguishing relations among objects in the decision system. Then, indiscernibility is the most important concept in the classical rough set and its extensions. Below, we present a formal definition of the indiscernibility used in our GIRM.

Definition 1. Generalized indiscernibility relation in GIRM. Given a decision system $DS = \langle U, C \cup D\rangle$ and $B \subseteq C$, $R_B \subseteq U \times U$ is a relation defined on U with respect to the attribute set B, for $\forall x, y \in U$, R_B is characterized by a membership function μ_{R_B} that is associated with each pair (x, y), and $\mu_{R_B}(x, y)$ represents the strength of the relation between x and y, $\mu_{R_B} \in [0, 1]$. If the relation R_B satisfies reflexivity: $\forall x \in U, \mu_{R_B}(x, x) = 1$, R_B is a generalized indiscernibility relation.

The generalized indiscernibility relation uses the weak restrain in terms of reflexivity, symmetry, and transitivity. The symmetry and transitivity can be relaxed, while the reflexivity cannot be relaxed because it is a minimal requirement for discernibility.

Definition 2. Granular structure in GIRM. In a decision system $DS = \langle U, C \cup D\rangle$, $U = \{x_1, x_2, \ldots, x_n\}$, $B \subseteq C$, R_B is a generalized indiscernibility relation. $\forall x_i, x_j \in U$, the granular structure GS_{R_B} determined by R_B is defined as

$$GS_{R_B} = \left\{ IG_{x_i}^{R_B} \mid IG_{x_i}^{R_B} = \frac{\mu_{R_B}(x_i, x_1)}{x_1} + \frac{\mu_{R_B}(x_i, x_2)}{x_2} + \ldots + \frac{\mu_{R_B}(x_i, x_n)}{x_n}, i = 1, 2, \ldots, n \right\}$$

$IG_{x_i}^{R_B}$ is an information granule of x_i, the degree of x_j belonging to the information granule $IG_{x_i}^{R_B}$ is denoted by $\mu_{IG_{x_i}^{R_B}}(x_j)$ and $\mu_{IG_{x_i}^{R_B}}(x_j) = \mu_{R_B}(x_i, x_j)$. If $\mu_{R_B}(x_i, x_j) \in \{0, 1\}$, $IG_{x_i}^{R_B}$ can be written as $\left\{x_j \mid x_j \in U \wedge \mu_{R_B}(x_i, x_j) = 1\right\}$.

In the above definition, "+" means the union of elements, and information granules are induced by the granulation strategies that are based on indiscernibility relations. The diversity of indiscernibility relations leads to the diversity of information granules. We illustrate it using an example.

Example 1. Table 1 is a decision system, $U = \{x_1, x_2, x_3, x_4, x_5, x_6\}$, $C = \{a, b, c\}$, $D = \{d\}$. Let $B = \{c\}$, R_B be the indiscernibility relation on U, and

$$\mu_{R_B}(x_i, x_j) = \begin{cases} 1, & c(x_i) = c(x_j) \\ 0, & c(x_i) \neq c(x_j) \end{cases}.$$

Table 1
Decision system for the example.

	a	b	c	d
x_1	1	1	1	0
x_2	1	2	3	2
x_3	2	3	2	0
x_4	3	1	2	1
x_5	1	1	1	1
x_6	1	2	3	2

$R_B^{'}$ is another indiscernibility relation, and $\mu_{R_B^{'}}(x_i, x_j) = 1 - \frac{|c(x_i) - c(x_j)|}{c_{max} - c_{min}}$. Although both R_B and $R_B^{'}$ are indiscernibility relations in GIRM, the information granules that are induced by them are different, e.g., $IG_{x_1}^{R_B} = \frac{1}{x_1} + \frac{0}{x_2} + \frac{0}{x_3} + \frac{0}{x_4} + \frac{1}{x_5} + \frac{0}{x_6}$, $IG_{x_1}^{R_B^{'}} = \frac{1}{x_1} + \frac{0}{x_2} + \frac{0.5}{x_3} + \frac{0.5}{x_4} + \frac{1}{x_5} + \frac{0}{x_6}$. As $IG_{x_1}^{R_B}$ is induced by a precise indiscernibility relation, it can be denoted by $IG_{x_1}^{R_B} = \{x_1, x_5\}$.

The information granules in GIRM are more generalized than equivalence classes in the classical rough set, and they can form a concept family to approximate the arbitrary subset of objects. Below, we give the corresponding definitions of the lower approximation and upper approximation in GIRM.

Definition 3. Lower approximation and upper approximation in GIRM. R_B is a generalized indiscernibility relation on the universe U. For any $X \subseteq U$, the lower and upper approximations of X are defined as follows:

$$\underline{apr}_{R_B} X = \left\{ (x, \mu_{\underline{apr}_{R_B} X}(x)) \in U \times [0,1] \mid \mu_{\underline{apr}_{R_B} X}(x) = \inf_{y \notin X \wedge y \in U} \left\{ 1 - \mu_{IG_x^{R_B}}(y) \right\} \right\}$$

$$\overline{apr}_{R_B} X = \left\{ (x, \mu_{\overline{apr}_{R_B} X}(x)) \in U \times [0,1] \mid \mu_{\overline{apr}_{R_B} X}(x) = \sup_{y \in X} \left\{ \mu_{IG_x^{R_B}}(y) \right\} \right\}$$

$\mu_{\underline{apr}_{R_B} X}(x)$ is the membership degree of x with respect to the lower approximation, and $\mu_{\overline{apr}_{R_B} X}(x)$ is the membership degree of x with respect to the upper approximation. As $\mu_{IG_x^{R_B}}(y) = \mu_{R_B}(x, y)$, $\mu_{IG_x^{R_B}}(y)$ can be replaced by $\mu_{R_B}(x, y)$ in Definition 3. Assuming that an ultrametric distance can be computed as $1 - \mu_{R_B}(x, y)$, then $\mu_{\underline{apr}_{R_B} X}(x)$ can be regarded as the ultrametric distance from x to the closest object that is not in X, $\mu_{\overline{apr}_{R_B} X}(x)$ is determined by the similarity degree between x and the closest object that is in X.

Example 2. In Table 1, the objects are divided into three subsets with the decision attribute d: $X_1 = \{x_1, x_3\}$, $X_2 = \{x_4, x_5\}$, $X_3 = \{x_2, x_6\}$. For object x_1 and indiscernibility relation $R_B^{'}$ in example 1, we calculate its membership degrees with respect to the lower approximation and upper approximation of each subset as follows: $\mu_{\underline{apr}_{R_B^{'}} X_1}(x_1) = \inf\{1, 0.5, 0, 1\} = 0$, $\mu_{\underline{apr}_{R_B^{'}} X_2}(x_1) = \inf\{0, 1, 0.5, 1\} = 0$, $\mu_{\underline{apr}_{R_B^{'}} X_3}(x_1) = \inf\{0, 0.5, 0.5, 0\} = 0$, $\mu_{\overline{apr}_{R_B^{'}} X_1}(x_1) = \sup\{1, 0.5\} = 1$, $\mu_{\overline{apr}_{R_B^{'}} X_2}(x_1) = \sup\{0.5, 1\} = 1$, $\mu_{\overline{apr}_{R_B^{'}} X_3}(x_1) = \sup\{0, 0\} = 0$.

Generally, the decision attribute set D induces crisp classification, denoted by U/D. In this paper, we focus only on this kind of decision system. Then, we define the decision positive region as follows.

Definition 4. Decision positive region in GIRM. Given a decision system $DS = \langle U, C \cup D \rangle$ and $B \subseteq C$, the positive region of D with respect to B in GIRM is $POS_B^{\#}(D) = \left\{ \bigcup_{x \in U} (x, \mu_{POS_B^{\#}(D)}(x)) \right\}$, $\mu_{POS_B^{\#}(D)}(x)$ is the membership degree of an object with respect to the positive region that is defined by

$$\mu_{POS_B^{\#}(D)}(x) = \sup_{X \in U/D} \mu_{\underline{apr}_{R_B} X}(x).$$

Theorem 1. *Given a decision system $DS = \langle U, C \cup D \rangle$ and $B \subseteq C$, R_B is the generalized indiscernibility relation, and objects are divided into crisp subsets $\{D_1, D_2, \ldots D_l\}$ with the decision attribute D. If $x \notin D_k$, then $\mu_{\underline{apr}_{R_B} D_k}(x) = 0$.*

Proof. Based on Definition 1, $\mu_{R_B}(x, x) = 1$. For $x \notin D_k$, from Definition 3, it follows that

$$\begin{aligned} \mu_{\underline{apr}_{R_B} D_k}(x) &= \inf_{y \notin D_k} \left\{ 1 - \mu_{IG_x^{R_B}}(y) \right\}, y \in U \\ &= \inf_{y \notin D_k} \{1 - \mu_{R_B}(x, y)\}, y \in U \\ &= \inf_{y \notin D_k} \{1 - \mu_{R_B}(x, x), 1 - \mu_{R_B}(x, y)\}, y \in U \wedge y \neq x \end{aligned}$$

$$= \inf_{y \notin D_k} \{1 - \mu_{R_B}(x,x), 1 - \mu_{R_B}(x,y)\}, y \in U \wedge y \neq x$$
$$= \inf_{y \notin D_k} \{0, 1 - \mu_{R_B}(x,y)\}, y \in U \wedge y \neq x$$
$$= 0.$$

□

The theorem indicates that the membership degree of x with respect to the decision positive region is closely determined by the membership degree of x with respect to the decision class to which x belongs. Therefore, $\mu_{POS_B^{\#}(D)}(x) = \mu_{\underline{apr}_{R_B} D_i}(x), x \in D_i$.

Another important issue in GIRM is the dependency. Based on the above theorem, the dependency function can be defined as follows.

Definition 5. Dependency function in GIRM. The dependency function of D with respect to B is defined as

$$DF(B|D) = \frac{\sum_{x \in U} \mu_{POS_B^{\#}(D)}(x)}{|U|}.$$

Example 3. In Table 1, let $B = \{c\}$, $D = \{d\}$. Considering the indiscernibility relation R_B' in example 1, $\mu_{POS_B^{\#}(D)}(x) = \mu_{\underline{apr}_{R_B'} D_i}(x), x \in D_i$. Hence, $\mu_{POS_B^{\#}(D)}(x_1) = 0$, $\mu_{POS_B^{\#}(D)}(x_2) = 0.5$, $\mu_{POS_B^{\#}(D)}(x_3) = 0$, $\mu_{POS_B^{\#}(D)}(x_4) = 0$, $\mu_{POS_B^{\#}(D)}(x_5) = 0$, $\mu_{POS_B^{\#}(D)}(x_6) = 0.5$. Then, the dependency degree of D with respect to B is $DF(B|D) = \frac{0 + 0.5 + 0 + 0 + 0 + 0.5}{6} = 0.17$.

The dependency function reflects the distinguishing capability of condition attributes. As $0 \leqslant \mu_{POS_B^{\#}(D)}(x) \leqslant 1$, we have $0 \leq DF(B|D) \leq 1$. Based on the essential idea of indiscernibility, the goal of the attribute reduction is to find a minimal attribute set that has the same distinguishing capability as all of condition attributes. Then the reduction model with generalized indiscernibility can be formally defined as follows.

Definition 6. Generalized reduction model. Given a decision system $DS = \langle U, C \cup D \rangle$ and $B \subseteq C$, $\forall a \in B$, B is the reduction if the following conditions are satisfied,

1. $DF(B|D) = DF(C|D)$;
2. $DF(B - a|D) < DF(C|D)$.

The first condition guarantees that the reduction has the same distinguishing capability as the whole condition attribute set, and the second condition guarantees that there is no redundant attribute in the reduction.

3.2. Unifying three typical rough set models with GIRM

Below, we employ three typical rough set models, i.e., the classical rough set, neighborhood rough set and fuzzy rough set, and we discuss their relation to our proposed GIRM. The discussion shows that these three typical rough set models can be incorporated into our GIRM.

3.2.1. Classical rough set model

Before discussing the relationship between the classical rough set [34] and our GIRM, we discuss several basic concepts regarding the classical rough set as follows [34].

Definition 7. Given a decision system $DS = \langle U, C \cup D \rangle$, for an attribute subset $B \subseteq A$, the indiscernibility relation in the classical rough set is defined by I_B :

$$I_B = \{(x,y) \in U \times U \mid \forall a \in B, a(x) = a(y)\}.$$

Obviously, I_B is an equivalence relation. The family of all equivalence classes of I_B can be denoted by U/I_B, or simply U/B.

Theorem 2. *Given a decision system $DS = \langle U, C \cup D \rangle$ and $B \subseteq C$, I_B is the indiscernibility relation in the classical rough set, and R_B is the generalized indiscernibility relation in GIRM. If R_B can be degenerated to I_B, then R_B should satisfy the following three necessary but insufficient conditions.*

C1: $\forall x, y \in U$, $\mu_{R_B}(x,y) = \mu_{R_B}(y,x)$,
C2: $\mu_{R_B} \in \{0,1\}$,
C3: $\forall x, y, z \in U$, $min(\mu_{R_B}(x,y), \mu_{R_B}(y,z)) \leqslant \mu_{R_B}(x,z)$.

Proof. From Definition 7, we obtain,

1. $\forall x, y \in U$, if $(x, y) \in I_B$, then $(y, x) \in I_B$; if $(x, y) \notin I_B$, then $(y, x) \notin I_B$, so $\mu_{I_B}(x,y) = \mu_{I_B}(y,x)$;
2. an arbitrary pair$(x, y) \in U \times U$, if $(x, y) \in I_B$, then $\mu_{I_B}(x,y) = 1$; otherwise, $\mu_{I_B}(x,y) = 0$;
3. $\forall x, y, z \in U$, if $(x, y) \in I_B$ and $(y, z) \in I_B$, then$(x, z) \in I_B$; otherwise, $(x, z) \notin I_B$. Therefore, $\mu_{I_B}(x,z) = min(\mu_{I_B}(x,y), \mu_{I_B}(y,z))$.

Thus, $C1$, $C2$, $C3$ are the necessary properties of I_B. If R_B can be degenerated to I_B, R_B needs to satisfy $C1$, $C2$, $C3$. □

In the classical rough set, $X \subseteq U$ can be approximated by the information granule $[x]_{I_B}$, $[x]_{I_B} = \{y | x I_B y, y \in U\}$, which is also an equivalence class of I_B. There are two approximations, i.e., the lower approximation $\underline{I_B} X$ and the upper approximation $\overline{I_B} X$:

$$\underline{I_B} X = \{[x]_{I_B} | [x]_{I_B} \subseteq X\}$$
$$\overline{I_B} X = \{[x]_{I_B} | [x]_{I_B} \cap X \neq \varnothing\}.$$

Then, the positive region of the decision is defined as:

$$POS_{I_B}(D) = \bigcup_{X \subseteq U/D} \underline{I_B} X.$$

The dependency function is:

$$\gamma_{I_B}(D) = \frac{|POS_{I_B}(D)|}{|U|}.$$

Theorem 3. *Given a decision system $DS = \langle U, C \cup D \rangle$ and $B \subseteq C$, I_B is the indiscernibility relation in the classical rough set. Then,* $DF(B|D) = \dfrac{\left|POS_{I_B}(D)\right|}{|U|}$.

Proof. Let $U/D = \{D_1, D_2, \ldots, D_l\}$. For $\forall x_i \in U$,

$$\left|POS_{I_B}(D)\right| \Leftrightarrow \mu_{POS_{I_B}(D)}(x_i) = \begin{cases} 1, & x_i \in POS_{I_B}(D) \\ 0, & x_i \notin POS_{I_B}(D) \end{cases}$$
$$\Leftrightarrow \mu_{POS_{I_B}(D)}(x_i) = \begin{cases} 1, & \exists D_k, [x_i]_{I_B} \subseteq D_k, k = 1, 2, \ldots, l \\ 0, & \forall D_k, [x_i]_{I_B} \nsubseteq D_k, k = 1, 2, \ldots l. \end{cases}$$

In GIRM, according to Definition 7, the membership function of I_B is $\mu_{I_B}(x_i, x_j) = \begin{cases} 1, & (x_i, x_j) \in I_B \\ 0, & (x_i, x_j) \notin I_B, \end{cases}$

If $\exists D_k, [x_i]_{I_B} \subseteq D_k$, then $\forall x_j \notin D_k$, $(x_i, x_j) \notin I_B \Rightarrow \mu_{I_B}(x_i, x_j) = 0 \Rightarrow$
$\mu_{\underline{apr}_{I_B}} D_k(x_i) = \inf_{x_j \notin D_k} \left\{1 - \mu_{I_B}(x_i, x_j)\right\} = 1 \Rightarrow \mu_{POS_B^{\#}(D)}(x_i) = 1.$

If $\forall D_k, [x_i]_{I_B} \nsubseteq D_k$, assuming $x_i \in D_t$, then $\exists x_j \notin D_t$, $(x_i, x_j) \in I_B \Rightarrow \mu_{I_B}(x_i, x_j) = 1$ and $\mu_{\underline{apr}_{I_B}} D_k(x_i) = \inf_{x_r \notin D_k} \left\{1 - \mu_{I_B}(x_i, x_j), 1 - \mu_{I_B}(x_i, x_r)\right\} = 0 \Rightarrow \mu_{POS_B^{\#}(D)}(x_i) = 0.$

Thus, $DF(B|D) = \dfrac{\sum_{x \in U} \mu_{POS_B^{\#}(D)}(x)}{|U|} = \dfrac{\left|POS_{I_B}(D)\right|}{|U|}$. □

Example 4. In Table 1, let $B = \{c\}$ and $D = \{d\}$, then $U/B = \{\{x_1, x_5\}, \{x_3, x_4\}, \{x_2, x_6\}.\}$, $U/D = \{\{x_1, x_3\}, \{x_2, x_6\}, \{x_4, x_5\}\}$, thus, $POS_{I_B}(D) = \{x_2, x_6\}$, $\gamma_{I_B}(D) = 1/3$. I_B is also the indiscernibility relation R_B in example 1. Considering R_B in GIRM, $\mu_{POS_B^{\#}(D)}(x_1) = 0$, $\mu_{POS_B^{\#}(D)}(x_2) = 1$, $\mu_{POS_B^{\#}(D)}(x_3) = 0$, $\mu_{POS_B^{\#}(D)}(x_4) = 0$, $\mu_{POS_B^{\#}(D)}(x_5) = 0$, $\mu_{POS_B^{\#}(D)}(x_6) = 1$, thus, $DF(B|D) = \dfrac{\sum_{x \in U} \mu_{POS_B^{\#}(D)}(x)}{|U|} = 1/3.$

3.2.2. Neighborhood rough set model

As the extension of classical rough set, the neighborhood relation based model [13] can process both numeric and discrete attributes. Since the neighborhood rough set model in paper [13] is representative and practical, we will discuss the relation between the neighborhood rough set [13] and our GIRM in this section. Several basic concepts in the neighborhood rough set are given firstly.

Definition 8. Given a decision system $DS = \langle U, C \cup D \rangle$, $U = \{x_1, x_2, \ldots, x_n\}$, $B \subseteq C$ and $\varepsilon \geq 0$, the neighborhood $\varepsilon_B(x_i)$ of x_i with respect to B is defined as $\varepsilon_B(x_i) = \left\{x_j \mid x_j \in U, f_B(x_i, x_j) \leqslant \varepsilon\right\}$, where $f_B(x_i, x_j)$ is a metric function that satisfies: for $\forall x_i$, x_j, $x_k \in U$,

(1) $f_B(x_i, x_j) \geq 0$, if $x_i = x_j$, $f_B(x_i, x_j) = 0$;
(2) $f_B(x_i, x_j) = f_B(x_j, x_i)$;
(3) $f_B(x_i, x_j) \leqslant f_B(x_i, x_k) + f_B(x_k, x_j)$.

$\varepsilon_B(x_i)$ is the neighborhood information granule, and the family of neighborhood granules forms the neighborhood granular structure, which covers the universe. A neighborhood relation N_B on U with respect to B can be presented as a relation matrix $M(N_B) = (r_{ij})_{n \times n}$ where

$$r_{ij} = \begin{cases} 1, & x_j \in \varepsilon_B(x_i) \\ 0, & otherwise \end{cases}$$

The neighborhood relation satisfies reflexivity and symmetry, and it draws the objects together in terms of distances[1]. If $\varepsilon = 0$, $\varepsilon_B(x)$ is an equivalence class and N_B is an equivalence relation.

Theorem 4. *Given a decision system $DS = \langle U, C \cup D\rangle$ and $B \subseteq C$, N_B is the neighborhood relation on U, and R_B is the generalized indiscernibility relation in GIRM. If R_B can be degenerated to N_B, then R_B should satisfy the following two necessary but insufficient conditions:*

C1: $\forall x, y, \in U$, $\mu_{R_B}(x, y) = \mu_{R_B}(y, x)$,

C2: $\mu_{R_B} \in \{0, 1\}$.

Proof. Similar to Theorem 1, it can be easily proven. □

Definition 9. Given a neighborhood decision system $\langle U, C \cup D, \varepsilon\rangle$, $X \subseteq U$, $B \subseteq C$, the lower and upper approximations of X in terms of N_B, are defined as [13]:

$$\underline{N_B}X = \{x_i \mid \varepsilon_B(x_i) \subseteq X, x_i \in U\}$$

$$\overline{N_B}X = \{x_i \mid \varepsilon_B(x_i) \cap X \neq \varnothing, x_i \in U\}$$

Definition 10. Given a neighborhood decision system $\langle U, C \cup D, \varepsilon\rangle$, $D_1, D_2, \ldots, D_n$ are the object subsets with decisions 1 to n. For $B \subseteq C$, the lower and upper approximations of decision D with respect to attributes B are defined as [13]:

$$\underline{N_B}D = \bigcup_{i=1}^{n} \underline{N_B}D_i$$

$$\overline{N_B}D = \bigcup_{i=1}^{n} \overline{N_B}D_i$$

The lower approximation of the decision is defined as the union of the lower approximation of each decision class. It is also called the positive region of the decision, and is denoted by $POS_{N_B}D$. Then, the dependency degree of D with respect to B in the neighborhood rough set model can be calculated as:

$$\gamma_{N_B}(D) = \frac{|POS_{N_B}(D)|}{|U|}$$

Theorem 5. *Given a neighborhood decision system $\langle U, C \cup D, \varepsilon\rangle$ and $B \subseteq C$, N_B is the neighborhood relation, then $DF(B|D) = \frac{\left|POS_{N_B}(D)\right|}{|U|}$.*

Proof. Similar to Theorem 3, it can be easily proven. □

3.2.3. Fuzzy rough set model

The concept of the fuzzy rough set was first proposed by Dubois and Prade [8], who constructed a pair of lower and upper approximation operators for fuzzy sets based on a fuzzy similarity relation. To generally apply the fuzzy rough set method, many extended versions [28,50,60] and relative applications [12,16,44,48] have been developed. In particular, to process hybrid data, Hu et al. [12] proposed a novel fuzzy rough set model with crisp lower and upper approximations, while Wang et al. [48]developed a generalized fuzzy rough set model using β-cut to define its lower and upper approximations. Liang et al. [49] discussed the relationships between the above three models. Hu's lower approximation is the 1-cut of Dubois's fuzzy lower approximation, and Hu's upper approximation is the strong 0-cut of Dubois's fuzzy upper approximation. Wang's fuzzy rough approximations are essentially equal to the β-cut of Dubois's fuzzy rough approximations. Without loss of generality, we select Dubois's fuzzy rough set model [8] as a representative in the following discussion. First, we introduce the central concept of the fuzzy similarity relation [57] as follows.

Definition 11. Let U be a nonempty universe. For $\forall x, y \in U$, $\mu_S(x, y)$ is the strength of fuzzy binary relation S between x and y. Then, a fuzzy binary relation S on U is called a fuzzy similarity relation if S satisfies [57] for $\forall x, y, z \in U$,

(1) *reflexivity:* $\mu_S(x, x) = 1$;

(2) *symmetry:* $\mu_S(x, y) = \mu_S(y, x)$;

(3) *transitivity:* $\mu_S(x, z) \geq \vee_y(\mu_S(x, y) \wedge \mu_S(y, z))$.

In a decision system $\langle U, C \cup D\rangle$, for every attribute $a \in A$, we can employ a fuzzy similarity relation S_a to measure the degree of similarity between each pair of objects with respect to a. For an attribute subset $B \subseteq C$, the fuzzy similarity relation induced by B is $S_B = \bigcap_{a \in B}\{S_a\}$.

Theorem 6. *Given a decision system $\langle U, C \cup D\rangle$, $B \subseteq C$ and S_B is the fuzzy similarity relation, and R_B is the generalized indiscernibility relation in GIRM. If R_B can be degenerated to S_B, then R_B should satisfy the following two necessary but insufficient conditions:*

[1] In this paper, we only discuss the metric function of Euclidean distance used in neighborhood rough set theory.

C1: $\forall x, y \in U$, $\mu_{R_B}(x, y) = \mu_{R_B}(y, x)$,
C3: $\forall x, y, z \in U$, $min(\mu_{R_B}(x, y), \mu_{R_B}(y, z)) \leqslant \mu_{R_B}(x, z)$.

Proof. Based on Definitions 1 and 11, it can be easily proven. □

According to the definition in [8], for $X \subseteq U$, the memberships of an object x with respect to the fuzzy lower and upper approximations of X are described as:

$$\mu_{\underline{S_B}X}(x) = \inf_{y \in U} \max\{1 - \mu_{S_B}(x, y), X(y)\}$$
$$\mu_{\overline{S_B}X}(x) = \sup_{y \in U} \min\{\mu_{S_B}(x, y), X(y)\}.$$

It is easy to prove that $\mu_{\underline{S_B}X}(x)$ and $\mu_{\overline{S_B}X}(x)$ are equal to $\mu_{\underline{apr_{R_B}}X}(x)$ and $\mu_{\overline{apr}_{R_B}X}(x)$ in GIRM, respectively. Based on the above definitions, the dependency function in the fuzzy rough set model can be calculated in the same way as in GIRM.

4. General quick reduction algorithms based on GIRM

There are various attribute reduction approaches in terms of rough set models, and many quick reduction algorithms have been also proposed. However these quick approaches may only work on specific indiscernibility relations. In this section, we will provide a series of systematic acceleration policies for the attribute reduction from the viewpoint of GIRM. Based on the generalized indiscernibility relation, we first present a basic generalized quick attribute reduction algorithm, and we then proposed a fast approach based on the monotonicity of the lower approximation in GIRM. In the reduction algorithms, we find that the largest computation cost of the attribute reduction is associated with computing the positive region, while the computation of the positive region is closely related to the indiscernibility relation. Based on the concept of the granular structure in our GIRM, the computation of the positive region can be regarded as involving the construction and iteration of the granular structure decoupled with the indiscernibility relation. Then, we further present a generalized positive region computing algorithm based on the granular structure in GIRM. Along the idea of computing the positive region with the granular structure, we also propose three fast positive region computing algorithms by accelerating the construction of the granular structures corresponding to those three typical rough set models, i.e., the classical rough set model, neighborhood rough set model and fuzzy rough set model.

4.1. Generalized quick reduction algorithm for GIRM

In attribute reduction algorithms [1,15,30,37], the dependency function is the most popular measure employed to represent the discernibility capability of attributes. Moreover, most quick attribute reduction algorithms are designed using the attribute significance based on the dependency function. In our GIRM, the significance of the attribute can be defined as follows.

Definition 12. The significance of an attribute in GIRM. Given a decision system $DS = \langle U, C \cup D\rangle$ and $B \subseteq C$, for $\forall a \in C - B$, the significance of an attribute a is

$$Sig(a, B, D) = DF(B \cup a|D) - DF(B|D).$$

Based on Definition 12 , we can propose a generalized attribute reduction algorithm in GIRM with a forward greedy search policy as follows.

In the loop of step 2, it needs to compute the significance of the attribute, which is equal to the change of the decision positive region. With respect to the computational complexity of Algorithm 1, we assume that there are k attributes included in the reduction and the temporal complexity involved in computing the positive region on U is denoted as $Pos(|U|)$, then the

Algorithm 1 Generalized forward greedy attribute reduction algorithm.

Input: $\langle U, C \cup D\rangle$;
Output: Reduction B.

step 1. $B \leftarrow \phi$;
step 2. for each $a_i \in C - B$
{ compute $Sig(a_i, B, D)$ };
step 3. select the attribute a_k satisfying $Sig(a_k, B, D) = max(Sig(a_i, B, D))$;
step 4. if $Sig(a_k, B, D) > 0$
$B \leftarrow B \cup a_k$, go to step 2;
else
return B;
step 5. return reduction B.

total computational time is $|C| \times Pos(|U|) + (|C|-1) \times Pos(|U|) + \ldots + (|C|-k) \times Pos(|U|) = Pos(|U|) \times \sum_{i=0}^{k}(|C|-i)$. Therefore, the temporal complexity of the attribute reduction algorithm depends mainly on the computation cost of the positive region. With respect to the computation of the $Pos(|U|)$, two strategies are employed to reduce the temporal complexity, the direct strategy involves reducing the computational domain of U, and the other one involves optimizing the computation of the positive region.

Based on the idea of the first strategy, we present a generalized quick forward greedy attribute reduction algorithm. Before presenting the fast algorithm, we first present the monotonicity theorem and its corollary in GIRM.

Theorem 7. *Given a decision system $\langle U, C \cup D\rangle$, $B_1 \subseteq B_2 \subseteq C$, R_{B_1} and R_{B_2} are generalized indiscernibility relations on U in GIRM. If $R_{B_2} \subseteq R_{B_1}$, then $\mu_{POS^{\#}_{B_1}(D)}(x) \leqslant \mu_{POS^{\#}_{B_2}(D)}(x)$ for $x \in U$.*

Proof. If $R_{B_2} \subseteq R_{B_1}$, then $\forall x, y \in U$. $\mu_{R_{B_2}}(x,y) \leqslant \mu_{R_{B_1}}(x,y)$. For an object x, assuming x belongs to the decision class D_i, we have $\inf_{y \in U - D_i}\left\{1 - \mu_{R_{B_2}}(x,y)\right\} \geq \inf_{y \in U - D_i}\left\{1 - \mu_{R_{B_1}}(x,y)\right\}$. That is, $\mu_{POS^{\#}_{B_1}(D)}(x) \leq \mu_{POS^{\#}_{B_2}(D)}(x)$. □

The essence of Theorem 7 is the monotonicity of the decision positive region with respect to the attributes. In practical applications, this is very important in the heuristic attribute reduction method because it ensures that the dependency function will increase monotonously as the attributes increase. Based on Theorem 7, for an arbitrary object $x \in U$, if $\mu_{POS^{\#}_{B_1}(D)}(x) = 1$, then $\mu_{POS^{\#}_{B_2}(D)}(x) = 1$. Therefore, we have the following corollary.

Corollary 1. *Given a decision system $DS = \langle U, C \cup D\rangle$, $B_1 \subseteq B_2 \subseteq C$, R_{B_1} and R_{B_2} are generalized indiscernibility relations on U in GIRM. If $R_{B_2} \subseteq R_{B_1}$, let $U' = \left\{x \mid \mu_{POS^{\#}_{B_1}(D)}(x) = 1, x \in U\right\}$, then $\sum_{i=1}^{|U|}\mu_{POS^{\#}_{B_2}(D)}(x_i) = \sum_{j=1}^{|U|-|U'|}\mu_{POS^{\#}_{B_1}(D)}(x_j) + |U'|$ for $x_i \in U$ and $x_j \in U - U'$.*

Proof. This can be easily proven from Theorem 7. □

Theorem 7 shows that the degree of an object necessary belonging to the decision positive region increases with the addition of attributes. Corollary 1 shows that once the membership degree of an object with respect to the decision positive region reached one, it will not change regardless of the subsequent addition of attributes. Then, for such objects, we do not need to compute their membership degrees with respect to the decision positive region when adding new attributes into the existing attribute subset. Thus, the generalized quick forward greedy attribute reduction algorithm is presented as follows.

In Algorithm 2, the domain U in each iteration to compute the positive region is gradually reduced. This will significantly improve the computational efficiency. Assuming that it will reduce $|U|/k$ objects in U when adding an attribute into the candidate attribute set, then the computational time of Algorithm 2 can be calculated as $|C| \times Pos(|U|) + (|C|-1) \times Pos(|U| \times \frac{k-1}{k}) + \ldots + (|C|-k) \times Pos(|U| \times \frac{1}{k})$, which is lower than that of Algorithm 1. Considering the second strategy to optimize the computation of the positive region, we present a generalized algorithm to compute the positive region based on the granular structure in GIRM, which is decomposed into two steps, i.e., constructing the granular structure and iterating the granular structure.

Algorithm 2 Generalized quick forward greedy attribute reduction algorithm.

Input: $\langle U, C \cup D\rangle$;
Output: Reduction B.

step 1. $B \leftarrow \phi$;
step 2. for each $a_i \in C - B$
{ compute $POS^{\#}_{B \cup a_i}(D)$ on U,
put the objects satisfying $\mu_{POS^{\#}_{B \cup a_i}(D)}(x) = 1$ into U'_{a_i}, $x \in U$,
compute $Sig(a_i, B, D)$ };
step 3. select the attribute a_k satisfying $Sig(a_k, B, D) = max(Sig(a_i, B, D))$;
step 4. $U \leftarrow U - U'_{a_k}$;
step 5. if $Sig(a_k, B, D) > 0$
$B \leftarrow B \cup a_k$, go to step 2;
else
return B;
step 6. return reduction B.

Theorem 8. *Given a decision system $DS = \langle U, C \cup D\rangle$ and $B \subseteq C$, R_B is the generalized discernibility relation in GIRM, and GS_{R_B} is the granular structure induced by R_B. If the time complexity to compute GS_{R_B} is represented by function $T(GS_{R_B})$, then the time complexity to compute $POS_B^{\#}(D)$ is $O(T(GS_{R_B}) + \sum |IG_{x_i}^{R_B}|)$.*

Proof. From Theorem 1, we know that $\mu_{POS_B^{\#}(D)}(x_i) = \mu_{\underline{apr}_{R_B} D_k}(x_i)$ where $x_i \in D_k$. In GIRM, computing $\mu_{\underline{apr}_{R_B} D_k}(x_i)$ is based on $IG_{x_i}^{R_B}$, and GS_{R_B} is constructed using $IG_{x_i}^{R_B}$ for $\forall x_i \in U$. That is, the decision positive region is hidden in the granular structure. Then there are two steps to compute the decision positive region, which is equal to computing each object's degree belonging to the decision positive region. The first step is to find the granular structure, and the second step is to iterate each information granule in the granular structure. Therefore, the time complexity to compute $POS_B^{\#}(D)$ is $O(T(GS_{R_B}) + \sum |IG_{x_i}^{R_B}|)$. □

Based on Theorem 8, there are two approaches to accelerating the positive region algorithm, namely optimizing the construction of the granular structure and the iteration of the granular structure. Below, we will present fast positive region computing algorithms from the above two perspectives with respect to three typical rough set models.

4.2. Fast positive region computing algorithm for classical rough set model

In the classical rough set, many fast approaches [10,27] have been proposed to compute the positive region in terms of the equivalence relation. After further analyzing those fast approaches, they all try to introduce sorting methods to the preprocessing of objects in order to accelerate the computation of the granular structures. In our previous work [30], we showed that the hash table can accurately represent the granular structure induced by the equivalence relation. Then, following the idea of Algorithm 3, we can present a fast positive region computing algorithm for the classical rough set model based on the hash table.

$B(x_j)$ is the numerical encoding of the object x_j with respect to the attribute subset B, and $hash(B(x_j))$ is the hash function that maps the object's numerical encoding into the hash item. All of the objects with the same encoding will be mapped into the same hash item. Based on Theorem 8, the temporal complexity of computing the positive region is composed of two components. With regard to the temporal complexity of constructing the granular structure in Algorithm 4, the hash table constructed by step 2 is the natural granular structure induced by the equivalence relation, thus, Algorithm 4 only needs to iterate $|U|$ times to construct the granular structure, and the temporal complexity of constructing the granular structure is $O(|U|)$. With respect to the temporal complexity of iterating the granular structure, the granular structure induced by

Algorithm 3 Generalized algorithm to compute positive region.

Input: $\langle U, C \cup D\rangle, B \subseteq C$;
Output: $POS_B^{\#}(D)$.

step 1. $GS_{R_B} \leftarrow \phi$;
step 2. for each $x_i \in U$ // Constructing the granular structure
{ compute $IG_{x_i}^{R_B}$, which is the information granule of x_i,
$GS_{R_B} \leftarrow GS_{R_B} \cup IG_{x_i}^{R_B}$ };
step 3. for each $IG_{x_i}^{R_B} \in GS_{R_B}$// Iterating the granular structure
{compute $\mu_{\underline{apr}_{R_B} D_k}(x_i)$ where $x_i \in D_k$};
step 4. return $POS_B^{\#}(D)$.

Algorithm 4 Fast positive region computing algorithm for classical rough model.

Input: $\langle U, C \cup D\rangle$, $B \subseteq C$;
Output: GS_{I_B} and $|POS_{I_B}(D)|$.

step 1. $GS_{I_B} \leftarrow \phi$, $Count = 0$, $IG_{x_i}^{I_B}.flag = true$, $IG_{x_i}^{I_B}.count = 0$;//GS_{I_B} is the hash table induced by I_B, $IG_{x_i}^{I_B}$ is the hash item
step 2. for each $x_j \in U$ // Constructing the granular structure induced by the equivalence relation
$\begin{cases} IG_{x_i}^{I_B} \leftarrow IG_{x_i}^{I_B} \cup hash(B(x_j)), \\ IG_{x_i}^{I_B}.count = IG_{x_i}^{I_B}.count + 1, \end{cases}$
if the decision of $IG_{x_i}^{I_B}$ is inconsistent, then $IG_{x_i}^{I_B}.flag = false$ };
step 3. for each $IG_{x_i}^{I_B} \in GS_{I_B}$// Iterating the granular structure induced by the equivalence relation
{ $Count = Count + IG_{x_i}^{I_B}.count$, where $IG_{x_i}^{I_B}.flag == true$ };
step 4. return GS_{I_B} and $Count$.//$Count$ is $|POS_{I_B}(D)|$

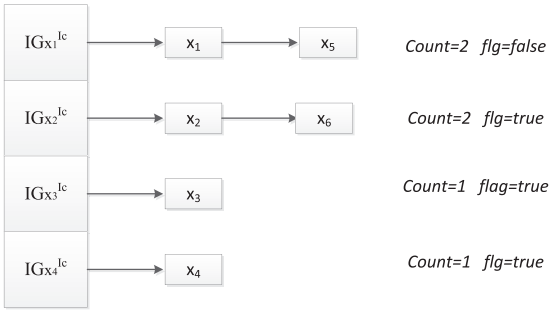

Fig. 1. The hash table induced by condition attribute set C.

the equivalence relation is a crisp partition on U, thus $\sum |IG_{x_i}^{I_B}| = |U|$, and the temporal complexity of iterating the granular structure is $O(|U|)$. In fact, the computation time required to iterate the granular structure can be further reduced. The positive region is related only to the information granule that all the objects in that information granule are consistent. Then we use $IG_{x_i}^{I_B}.flag$ to represent the consistency of the information granule, and step 3 only needs to process the information granules with $IG_{x_i}^{I_B}.flag = true$. Therefore the practical times required to iterate the granular structure will be much less than $|U|$.

Here, we will use an example to provide detailed illustrative process of Algorithm 4.

Example 5. In Table 1, $C = \{a, b, c\}$, and we then obtain the numerical encoding of each object with C as follows: $C(x_1) = 111$, $C(x_2) = 123$, $C(x_3) = 232$, $C(x_4) = 311$, $C(x_5) = 111$, $C(x_6) = 123$. After all of the objects have been mapped into hash items according to the hash function(step 2), the hash table is as shown in Fig. 1, which is also the granular structure GS_{I_C}. During the process of constructing the hash table, the number of objects in each information granule can be counted, and the consistency of the information granule can be checked. Then, by iterating the hash table(step 3) with the aid of $IG_{x_i}^{I_C}.count$ and $IG_{x_i}^{I_C}.flag$, we have $POS_{I_C}(D) = \{\{x_2, x_6\}, \{x_3\}, \{x_4\}\}$ and $|POS_{I_C}(D)| = 4$.

4.3. Fast positive region computing algorithm for neighborhood rough set model

In the neighborhood rough set [17], objects in the same neighborhood information granule are drawn together by the Euclidean distance. As the neighborhood relation can not satisfy the $C3$(transitivity), the neighborhood granular structure is a cover, but not a partition, on the universe. Thus, the neighborhood granular structure cannot be constructed by the hash function in Algorithm 4. Generally, it needs to iterate all objects in U in order to obtain the neighborhood information granule of each object, and the computation cost required to obtain the neighborhood granular structure will thus approximate to $O(|U|^2)$. Using the sorting technique, the time complexity can be reduced. In our previous work [54], we used hash buckets to approximately represent the neighborhood granular structure. Then following the idea of Algorithm 3, we can present a fast positive region computing algorithm for the neighborhood rough set model based on the hash buckets. Before presenting the fast algorithm, we introduce below the definition and the theorem related to the hash buckets.

Definition 13. In a neighborhood decision system $NDS = \langle U, C \cup D, \varepsilon \rangle$, x_0 is a special object constructed from U, where $\forall a \in C$, $a(x_0) = min\{a(x_i)\}$, $x_i \in U$. Then, the objects in U can be divided into finite buckets $B_0, \ldots, B_k$:

$B_k = \{x_i \mid x_i \in U and \lceil f(x_0, x_i)/\epsilon \rceil = k\}$.

The bucket can be regarded as a hash function mapping with the distance from x_i to x_0. As a result, the distribution of objects can be illustrated in Fig. 2. Based on the definition of buckets, we obtain the following theorem.

Theorem 9. *In a neighborhood decision system $NDS = \langle U, C \cup D, \epsilon \rangle$, $B_0, \ldots, B_k$ are the buckets, then $\forall x_i \in B_q (q = 1, 2, 3, \ldots, k, k-1)$, and the neighborhood elements of x_i are only contained in B_{q-1}, B_q, B_{q+1}. If $x_i \in B_0$, then the neighborhood elements are only contained in B_0, B_1. If $x_i \in B_k$, then the neighborhood elements of x_i are only contained in B_{k-1}, B_k.*

The detailed proof is in [54]. As shown in Fig. 2, the neighborhood set of x_i can be viewed as a hypersphere with a center of x_i and a radius of ε. The elements in x_i's neighborhood set are only located at the buckets B_{k-1}, B_k, B_{k+1}, where B_k is the bucket that contains x_i, B_{k-1}, and B_{k+1} are x_i's adjacent buckets. Based on Theorem 9, the neighborhood granular structure can be constructed quickly by iterating fewer objects. The fast algorithm that is employed to compute the positive region in the neighborhood rough set is presented as follows. Step 2 maps all objects into their corresponding hash buckets with the temporal complexity of $O(|U|)$. Assuming there are k buckets, step 3 constructs granular structure GS_{N_B} with computation cost $|B_0| \times |B_1| + |B_{k-1}| \times |B_k| + \sum_{i=1}^{k-1} |B_i| \times (|B_{i-1}| + |B_i| + B_{i+1})$, which is related to the distribution of objects. If the objects are averaged into the buckets, the computational times are $\frac{|U|^2}{k} \times 2 + (|U| - \frac{|U|}{k} \times 2) \times (\frac{|U|}{k} \times 3)$. In most cases, k will be close to $|U|$, then, the computational times of step 3 can approximate to $\sum_{i=1}^{n} |IG_{x_i}^{N_B}|$, that is, the temporal complexity of

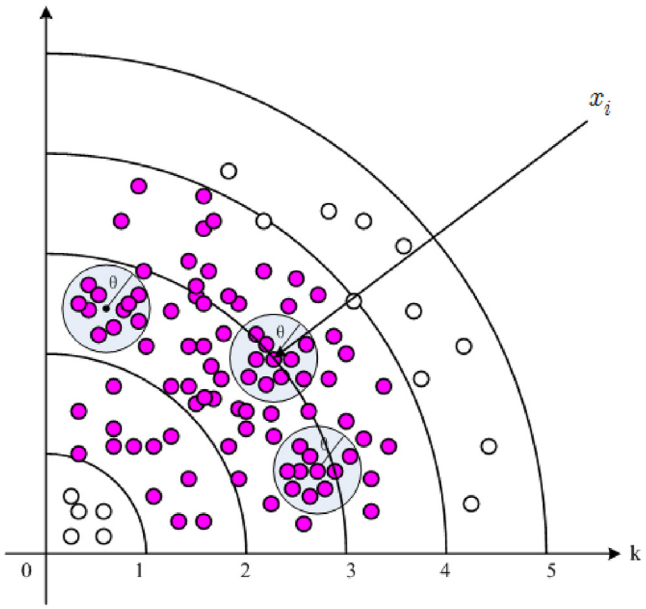

Fig. 2. The distribution of the buckets after mapping [54].

step 3 can be approximated to $O(c|U|)$. Here, $|IG_{x_i}^{N_B}|$ is the number of objects in a neighborhood information granule $\varepsilon(x_i)$, c is a constant. Step 4 iterates each neighborhood information granule and tests whether or not the objects in it have the same decision. If they have the same decision, x_i is in the decision positive region. The temporal complexity of step 4 is $O|U|$. So the total computational complexity of Algorithm 5 can approximate to $O(c|U|)$.

Algorithm 5 Fast positive region computing algorithm for neighborhood rough set model.

Input: $\langle U, C \cup D\rangle$, $B \subseteq C$, ε;
Output: GS_{N_B} and $POS_{N_B}(D)$.

step 1. $GS_{N_B} \leftarrow \phi$, $POS_{N_B}(D) \leftarrow \phi$;
step 2. for each $x_i \in U$
{ map each x_i into its corresponding bucket B_k by hash function };
step 3. for each $x_i \in U$//Constructing the neighborhood granular structure
{ compute $IG_{x_i}^{N_B}$ in $B_{k-1} \cup B_k \cup B_{k+1}$,//$IG_{x_i}^{N_B}$ is $\epsilon(x_i)$
$GS_{N_B} \leftarrow GS_{N_B} \cup IG_{x_i}^{N_B}$ };
step 4. for each $IG_{x_i}^{N_B} \in GS_{N_B}$ $(x_i \in U)$//Iterating the neighborhood granular structure
{ if the decision of $IG_{x_i}^{N_B}$ is consistent, then $POS_{N_B}(D) \leftarrow POS_{N_B}(D) \cup x_i$ };
step 5. return GS_{N_B} and $POS_{N_B}(D)$.

4.4. Fast positive region computing algorithm for fuzzy rough set theory

Attribute reduction using fuzzy rough sets was first proposed by Jenson and Shen [19], and there are also many improved approaches [1,12,21,23]. Generally, to construct the fuzzy information granule of x_i we needs to compute the similarity degrees between x_i and all the other objects. In Section 3, we observe that the fuzzy attribute reduction can be transformed to our GIRM when employing Dubois's fuzzy rough set model. In GIRM, the calculation for the membership degree of the object x_i belonging to the decision positive region concerns only the objects that are not in the x_i's decision class, reducing the computation required in the step to calculate fuzzy information granules. Based on this idea, we present a fast positive region computing algorithm for fuzzy rough set as follows.

Step 2 needs to calculate $|U|$ times to obtain decision classes U/D, this step aims to quickly search the objects that have the conflict decisions with the object x_i. Generally, computing fuzzy granular structure depends on the relation matrix $M(S_B)$ with the computation cost of $|U|^2$. Based on our GIRM, assuming the number of decision classes is t, the calculation of the relation matrix $M(S_B)$ can be simplified to the calculation of the relation matrix $M'(S_B)$. Both $M(S_B)$ and $M'(S_B)$ are represented in the following, r_{ij} is the similarity degree $\mu_{S_B}(x_i, x_j)$ between x_i and x_j, D_{mn} is the block of relation matrix

Algorithm 6 Fast positive region computing algorithm for fuzzy rough set model.

Input: $\langle U, C \cup D\rangle$, $B \subseteq C$;
Output: $POS_{S_B}(D)$.

step 1. $GS_{S_B} \leftarrow \phi$;
step 2. for each $x_i \in U$
{ Mapping x_i in decision classes $D_1, D_2, \ldots, D_t$ };
step 3. for each $x_i \in U$ // Constructing the fuzzy granular structure
{ compute fuzzy information granular $IG_{x_i}^{S_B}$ on $U - D_k$ where $x_i \in D_k$,
$GS_{S_B} \leftarrow GS_{S_B} \cup IG_{x_i}^{S_B}$ };
step 4. for each $IG_{x_i}^{S_B} \in GS_{S_B}$// Iterating the fuzzy granular structure
{ compute the membership degree of x_i belonging to the decision positive region $\mu_{POS_{S_B}} D(x_i)$ };
step 5. return $POS_{S_B}(D)$.

Table 2
Data description.

	Data	Object number	Attribute number	Attribute category	Classies
1	Balance scale	625	4	Symbolic	4
2	Car evaluation	1728	6	Symbolic	4
3	Chess	3196	36	Symbolic	2
4	Connect-4	67,557	42	Symbolic	2
5	Landsat	4435	36	Numerical	7
6	Magic gamma telescope (Mgt)	19,020	11	Numerical	2
7	Mushroom	8124	22	Symbolic	2
8	Nursery	12,960	8	Numerical	5
9	Page blocks	5473	10	Numerical	5
10	Pendigits	10,992	17	Numerical	10
11	Segmentation	2310	19	Numerical	7
12	Vehicle	846	18	Numerical	4

containing the similarity degree $\mu_{S_B}(x_i, x_j)$ for $x_i \in D_m$ and $x_j \in D_n$. Therefore, step 3 constructs the fuzzy granular structure with the computational times of $\sum_{i=1}^{t} |D_i| \times (|U| - |D_i|)$, which is related to the distribution of objects. If the objects are averagely divided into the decision classes, the computational times of step 3 are $\frac{t-1}{t} \times |U|^2$. Comparing to the computation cost $|U|^2$ in the previous work, step 3 in Algorithm 6 can reduce much time, and the fewer the number of decision classes is, the more time it saves. Step 4 iterates the fuzzy granular structure to compute the membership degree of each object belonging to the decision positive region, the computation cost of step 4 is the same as step 3. So Algorithm 6 can quickly compute the decision positive region in fuzzy rough set model.

$$\begin{bmatrix} IG_{x_1}^{S_B} \\ IG_{x_2}^{S_B} \\ \vdots \\ IG_{x_n}^{S_B} \end{bmatrix} = M(S_B) = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{bmatrix} \Rightarrow M'(S_B) = \begin{bmatrix} \text{ø} & D_{12} & \cdots & D_{1t} \\ D_{21} & \text{ø} & \cdots & D_{2t} \\ \vdots & \vdots & \text{ø} & \vdots \\ D_{t1} & D_{t2} & \cdots & \text{ø} \end{bmatrix}$$

5. Experiments

In the previous section, we discussed the attribute reduction algorithms and the acceleration policies. In this section, we discuss the comparable experiments to evaluate the correctness and efficiency of algorithms with acceleration policies on three typical rough set models. We used 12 datasets from UCI, and the details are shown in Table 2.

We run all of the algorithms on the same software and hardware platforms (CPU:Intel (R) core (TM)i7-4770 @3.40 Hz; RAM: 8.00 GB; Windows 7; Python 2.7). We execute the same algorithm on the same input dataset over 10 times, and we calculated the average execution time.

5.1. Experiments using classical rough set model

We compare four algorithms based on the classical rough set model. Algorithm C [18] is the generalized forward greedy classical attribute reduction algorithm without any acceleration policies. Algorithm C1 is the quick attribute reduction algorithm that uses the first acceleration policy, reducing the computational domain(F2DARPRS in [13]). Algorithm C2 employs the second acceleration policy, optimizing the computation of the classical granular structure by Algorithm 4, without using the first acceleration policy. Algorithm C12 uses both the first and second acceleration policies. We employ six datasets from

Table 3
The reduction results of four algorithms in classical rough set model.

Dataset	C	C1	C2	C12
Balance scale	1	1	1	1
Car evaluation	4,6	4,6	4,6	4,6
Chess	21,10,29,14,28	21,10,29,14,28	21,10,29,14,28	21,10,29,14,28
Connect-4	1,36,7,13,31,25,19,37,2, 38,14,8,20,15,32,21,26,9, 3,39,33,16,27,10,22,4,28, 34,41,17,11,23,5,29	1,36,7,13,31,25,19,37,2, 38,14,8,20,15,32,21,26,9, 3,39,33,16,27,10,22,4,28, 34,41,17,11,23,5,29	1,36,7,13,31,25,19,37,2, 38,14,8,20,15,32,21,26,9, 3,39,33,16,27,10,22,4,28, 34,41,17,11,23,5,29	1,36,7,13,31,25,19,37,2, 38,14,8,20,15,32,21,26,9, 3,39,33,16,27,10,22,4,28, 34,41,17,11,23,5,29
Mushroom	5,19,8,12,3	5,19,8,12,3	5,19,8,12,3	5,19,8,12,3
Nursery	8	8	8	8

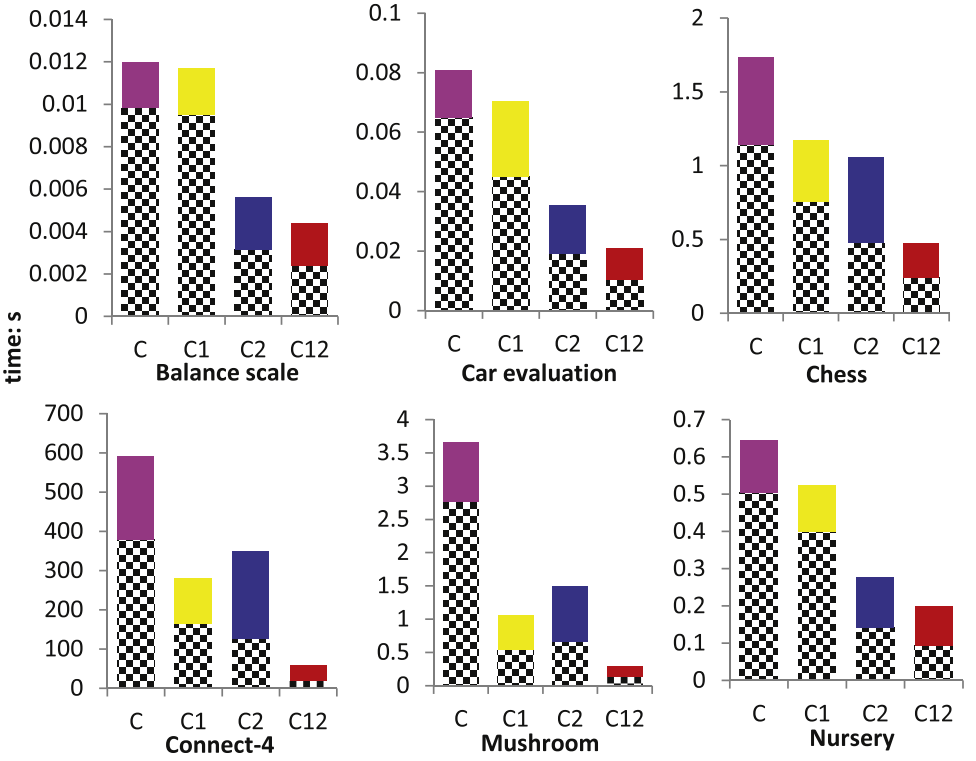

Fig. 3. The reduction time and positive region computational time in classical rough set model. The Y-axis is the computational time (S).

Table 2, i.e., *balance scale, car evaluation, chess, connect-4, mushroom,* and *nursery*. Those data are symbolic with the numbers from 625 to 67,557.

We first evaluate the correctness of the above four algorithms. The reduction results obtained for each dataset are shown in Table 3. On the same dataset, all four algorithms can output exactly the same reduction result.

In addition, we perform the comparable efficiency evaluation experiments on those datasets in Table 3. The temporal computation results of those four algorithms are shown in Fig. 3, where the shadow of each bar represents the total positive region computation time in the reduction, and the bar represents the whole reduction time. The high proportion of the shadow to the bar indicates that the largest portion of the reduction is computing the positive region. Considering the time for computing positive region, C12 always achieves the best performance in those four algorithms, C2 always performs better than C1 except on the *mushroom* dataset, while C1 performs better than C2 only on this dataset. After further analysis, we find that only the *mushroom* dataset leaves 25% of all objects continue to be proceeded after the first attribute is selected into the reduction subset, which is not common in most practical applications. Therefore, C1 has the advantage on this dataset. As shown in Fig. 3, the second acceleration policy may be more generalized than the first acceleration policy.

Further, we measure the computation time of the positive region based on all attributes of the dataset(denoted as the single positive region computational time), with the number of objects increasing in C and C2. These two algorithms were executed on exactly the same data objects. The results in Fig. 4 show that C2 is much faster than C, especially on the large datasets such as *connect-4, mushroom*. Then, the results indicate that Algorithm 4 (the second proposed policy) can promote the efficiency to compute the classical granular structure.

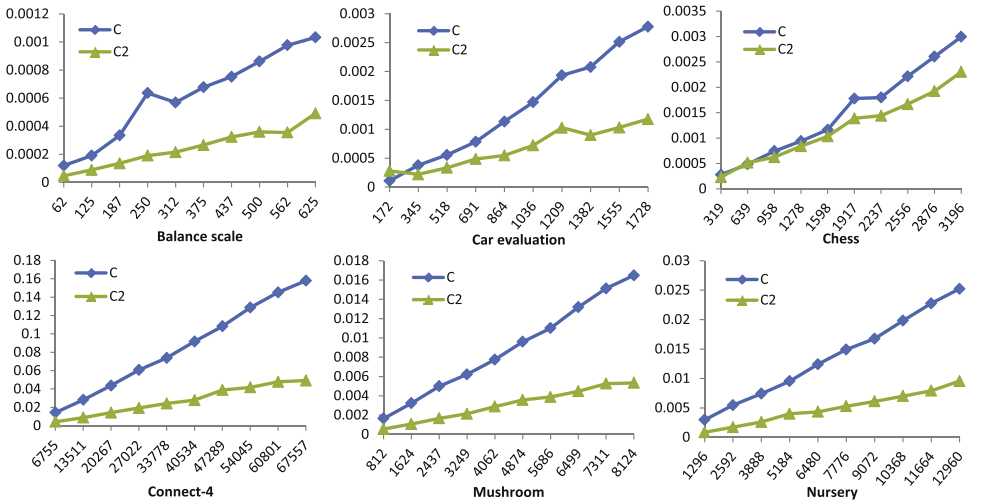

Fig. 4. The single positive region computational time in classical rough set model. The X-axis represents the number of objects and the Y-axis is the computational time (S).

Table 4
The reduction results of four algorithms in neighborhood rough set model.

Dataset	ε	N	N1	N2	N12
Landsat	0.05	20,32,29,2,21,5,26,23,11, 30,3,1,9,6,7	20,32,29,2,21,5,26,23,11, 30,3,1,9,6,7	20,32,29,2,21,5,26,23,11, 30,3,1,9,6,7	20,32,29,2,21,5,26,23,11, 30,3,1,9,6,7
Mgt	0.3	7,8,6,9,10,4,1,2,3,5	7,8,6,9,10,4,1,2,3,5	7,8,6,9,10,4,1,2,3,5	7,8,6,9,10,4,1,2,3,5
Page blocks	0.1	8,6,10,2,5,4,1,9,3,7	8,6,10,2,5,4,1,9,3,7	8,6,10,2,5,4,1,9,3,7	8,6,10,2,5,4,1,9,3,7
Pendigits	0.03	4,16,5,10,9,7,1	4,16,5,10,9,7,1	4,16,5,10,9,7,1	4,16,5,10,9,7,1
Segmentation	0.12	19,17,2,1,14,18,16,6,8,4	19,17,2,1,14,18,16,6,8,4	19,17,2,1,14,18,16,6,8,4	19,17,2,1,14,18,16,6,8,4
Vehicle	0.16	9,19,15,18,16,13,1,4,11	9,19,15,18,16,13,1,4,11	9,19,15,18,16,13,1,4,11	9,19,15,18,16,13,1,4,11

5.2. Experiments using neighborhood rough set model

We evaluate the correctness and computational efficiency of four attribute reduction algorithms based on the neighborhood rough set model. Algorithm N is the generalized forward greedy neighborhood attribute reduction algorithm(NFARNRS in [13]). Algorithm N1 uses the first acceleration policy reducing the computational domain(F2HARNRS in [13]). Algorithm N2 uses the second acceleration policy, optimizing the computation of the neighborhood granular structure obtained by Algorithm 5, without using the first acceleration policy. Algorithm N12 uses both the first and second acceleration policies. There are six datasets, which consist of numerical attributes from Table 2, i.e., *landsat, magic gemma telescope(mgt), page blocks, pendigits, segmentation,* and *vehicle*, and the number of objects in those datasets range from 846 to 10,992.

We compared the reduction results of the above four algorithms with the same ε value on the same dataset, and we randomly chose the ε value. The reduction results are shown in Table 4, which shows that these four algorithms output exactly the same reduction results.

Then, we evaluate the efficiency of those four algorithms. Different from classical rough set model, the positive region computational time based on the neighborhood rough set model is almost equal to the reduction computational time. So Fig. 5 only draws the computation time of the reduction, and is added to the legend to denote the ratio of the computation time between the positive region and the reduction. The results in Fig. 5 show that N12 achieves the best performance of all the datasets, specifically, N12 is 55 times faster than N in the *segmentation* dataset, and N12 is 90 times faster than N in the *landsat* dataset. Moreover, both N1 and N2 can achieve better performances than N. Fig. 5 shows that the first acceleration policy and the fast neighborhood granular computing method in Algorithm 5(the second policy) are much more effective strategies to reduce the reduction time in the neighborhood rough set model.

In the experiment described below, we increase the number of objects and measure the single positive region computational time in N and N2. We executed the two algorithms on exactly the same data objects. The ε value for each dataset was the same as that in the Table 4. The execution time obtained for various objects is shown in Fig. 6. This figure shows that N2 always performs better than N1, and that the single positive region computation costs for N increase significantly, while the costs for N2 increase much slower as the number of dataset is increased. We observe that the difference in the growth rates between N and N2 is less significant on *vehicle* dataset compared to the other five datasets. Compared to the number of the datasets for the original object, there are significantly fewer objects in the *vehicle*, so on this dataset, N2 has

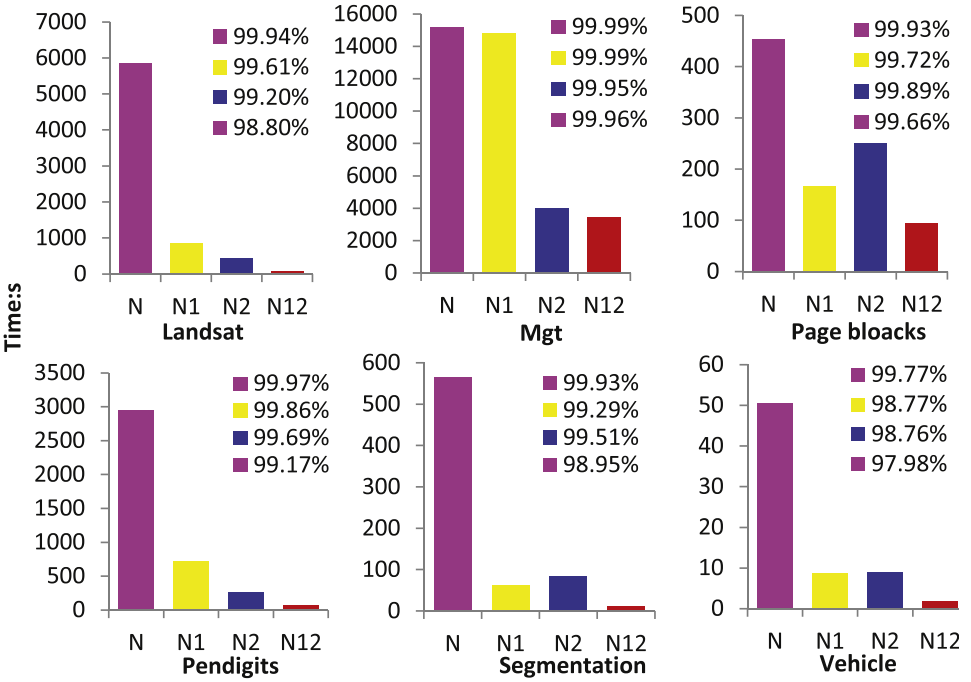

Fig. 5. The reduction computational time in neighborhood rough set model. The Y-axis represents the computational time (S), while the legend denotes the ratio of the computational time between the positive region and the reduction.

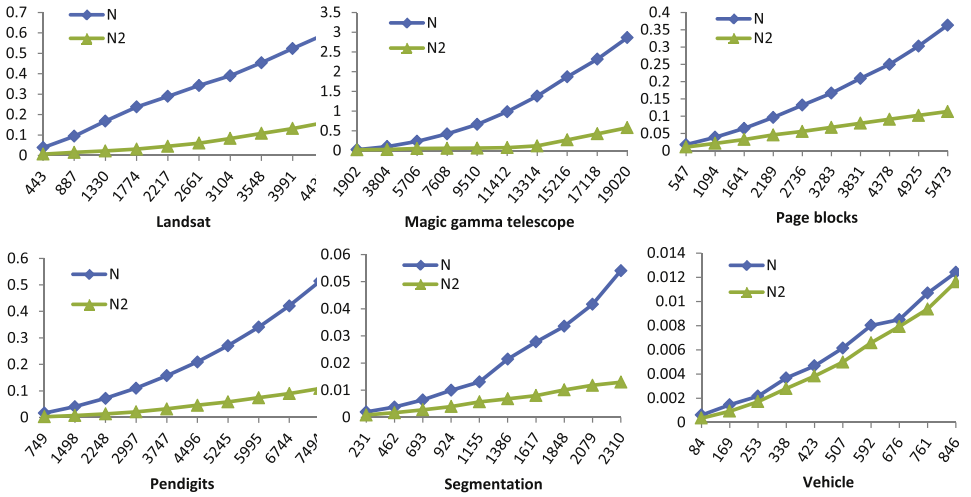

Fig. 6. The single positive region computational time in neighborhood rough set model. The X-axis is the number of objects and the Y-axis is the computational time (S).

slight advantage. Fig. 6 shows that the second acceleration policy employed in N2 can promote efficiency to compute the positive region, especially when there is a large number of objects.

Fig. 6 also shows that the increase in the cost to compute the single positive region is nearly linear in N2. For further study the efficiency of the second acceleration policy employed in N2, we perform an additional experiment to compare the relation between the theoretical time complexity and the real running time required to compute the single positive region in N2. Let B be the original attribute set. According to Theorem 8, the time complexity to compute the single positive region is $O(T(GS_{N_B}) + \sum |IG_{x_i}^{N_B}|)$, where $T(GS_{N_B})$ is the computation to construct the neighborhood granular structure and $\sum |IG_{x_i}^{R_B}|$ is the computation to traverse the neighborhood granular structure. The basic operations to construct GS_{N_B} are mapping object x_i in buckets and computing the distance between x_i and other object. We can use the number of the basic operations to calculate $T(GS_{N_B})$ for the construction of GS_{N_B}, which is equal to $|U| + \sum_{i=1}^{|U|}(|B_{j-1}| + |B_j| + |B_{j+1}|)$, where x_i

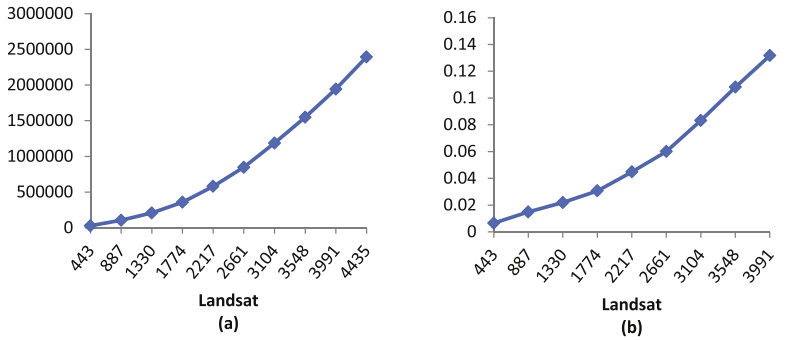

Fig. 7. The theoretical time complexity and the real running time required to compute the single positive region in neighborhood rough set model. The Y-axis in (a) is the number of basic operations(constructing the neighborhood structure) required to compute the single positive region, and the Y-axis in (b) is the real running time required (S) to compute the single positive region.

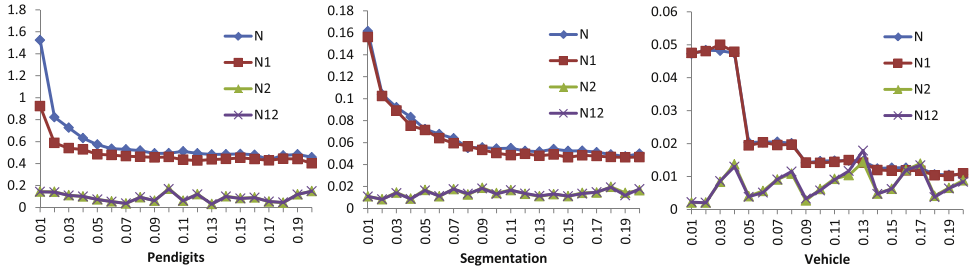

Fig. 8. The single positive region computational time for various values of ε in N2. The X-axis is the value of ε, and the Y-axis is the computational time (S).

$\in B_j$, B_j is the hash bucket. As $\sum |IG_{x_i}^{N_B}|$ is smaller than $T(GS_{N_B})$, $T(GS_{N_B})$ can be used to evaluate the time complexity to compute the positive region. The result for the *landsat* dataset is shown in Fig. 7(a). The vertical axis in Fig. 7(a) represents the number of basic operations(constructing the neighborhood structure) that are required to compute the single positive region in N2. Fig. 7(b) shows the corresponding real running time. By comparing Fig. 7(a) and (b), we find that the curves are very similar, which can further verify our theoretical analysis on the time complexity.

The computation of the neighborhood granular structures is sensitive to the setting of ε, so we choose three datasets, *pendigits, segmentation,* and *vehicle*, to evaluate the single positive region computation performance for various values of ε, which ranges from 0 to 0.2. There are 10,992, 2310, and 846 objects in *pendigits, segmentation,* and *vehicle*, respectively. The results in Fig. 8 show that N2 and N12 perform much better than N and N1, respectively.

5.3. Experiments using fuzzy rough set model

In this section, we evaluate the correctness and computational efficiency of four attribute reduction algorithms based on the fuzzy rough set model. Algorithm F is the generalized forward greedy fuzzy attribute reduction algorithm [23]. Algorithm F1 uses the first acceleration policy [38]. Algorithm F2 uses the second acceleration policy, which optimizes the computation of the fuzzy granular structure with Algorithm 6. Algorithm F12 uses both acceleration policies. The datasets are the same as those used in the neighborhood rough set model.

In the first experiment, the value of the fuzzy similarity degree $\mu_{S_a(x_i,x_j)}$ between objects x_i and x_j with respect to numerical attribute a is computed as

$$\mu_{S_a}(x_i, x_j) = 1 - \frac{a(x_i) - a(x_j)}{a_{max} - a_{min}}. \tag{1}$$

Table 5 shows the reduction results of four algorithms for each dataset. The results of the attribute reductions are exactly the same on the same dataset.

In the computational efficiency experiment, similar to the neighborhood rough set model, Fig. 9 displays only the results of the reduction time, and the legend denotes the ratio of the computational time between the positive region and the reduction. The results show that the computational time of F1 is nearly the same as that of F2. The reason is that using the formula (1) to compute the fuzzy similarity degree between objects, the degree of objects belonging to the positive region hardly reaches one. Then, the first acceleration policy does not perform well. In Fig. 9, it is easy to see that F2 is always faster than F. In particular, for large dataset, F2 results in a time savings of 2828s compared to F on the *mgt* dataset and F2

Table 5
The reduction results of four algorithms using formula (1) in fuzzy rough set model.

Dataset	F	F1	F2	F12
Landsat	20,5,22,33,13,34,36,4,25, 15,35,21,6,26,27,7,8,17, 14,10,30,28,23,2,12,1	20,5,22,33,13,34,36,4,25, 15,35,21,6,26,27,7,8,17, 14,10,30,28,23,2,12,1	20,5,22,33,13,34,36,4,25, 15,35,21,6,26,27,7,8,17, 14,10,30,28,23,2,12,1	20,5,22,33,13,34,36,4,25, 15,35,21,6,26,27,7,8,17, 14,10,30,28,23,2,12,1
Mgt	7,9,10,4,8,6,5,3,1,2	7,9,10,4,8,6,5,3,1,2	7,9,10,4,8,6,5,3,1,2	7,9,10,4,8,6,5,3,1,2
Page blocks	6,5,2,9,4,1,10,8,3,7	6,5,2,9,4,1,10,8,3,7	6,5,2,9,4,1,10,8,3,7	6,5,2,9,4,1,10,8,3,7
Pendigits	4,8,19,15,1,16,5,7,10, 13,3,12,2,11,14,6	4,8,19,15,1,16,5,7,10, 13,3,12,2,11,14,6	4,8,19,15,1,16,5,7,10, 13,3,12,2,11,14,6	4,8,19,15,1,16,5,7,10, 13,3,12,2,11,14,6
Segmentation	19,14,11,2,1,16,18,4,5, 6,8,13,7	19,14,11,2,1,16,18,4,5, 6,8,13,7	19,14,11,2,1,16,18,4,5, 6,8,13,7	19,14,11,2,1,16,18,4,5, 6,8,13,7
Vehicle	12,18,15,10,1,3,8,13,4, 17,2,6,9,5,14	12,18,15,10,1,3,8,13,4, 17,2,6,9,5,14	12,18,15,10,1,3,8,13,4, 17,2,6,9,5,14	12,18,15,10,1,3,8,13,4, 17,2,6,9,5,14

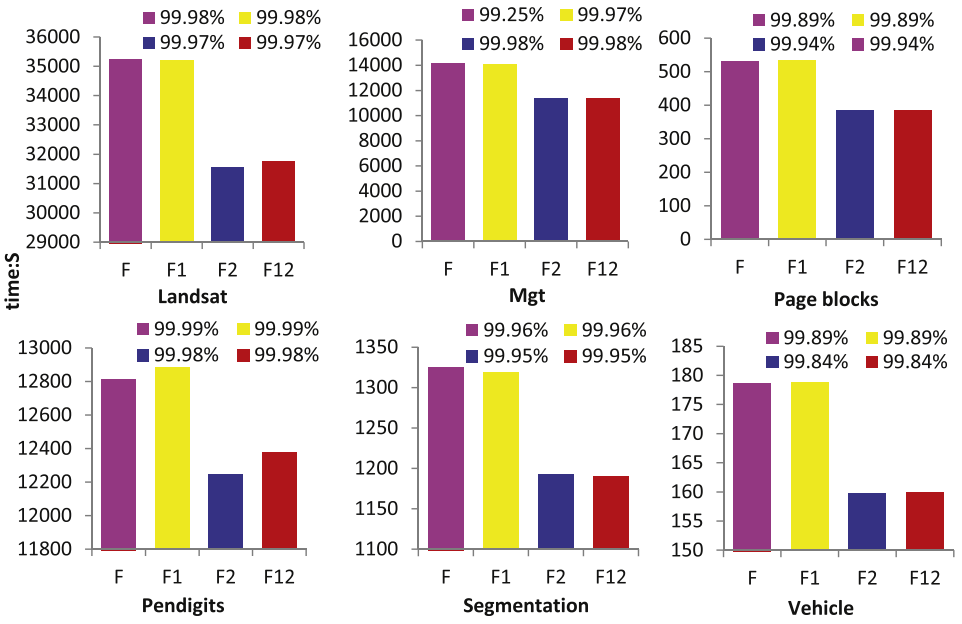

Fig. 9. The reduction computational time in fuzzy rough set model. The Y-axis represents the computational time (S), the legend denotes the ratio of the computational time between the positive region and the reduction.

results in a time savings of 3567s compared to F on the *landsat* dataset. Thus, the second policy can effectively reduce the computational time in fuzzy rough set model.

We also evaluate the single positive region computational time in F and F2 as the number of objects increases, and the four algorithms are executed on exactly the same data objects. We present the results in Fig. 10, which shows that F2 is always faster than F. Therefore, the proposed Algorithm 6(the second policy) can quickly compute the fuzzy granular structure.

As the degree of objects belonging to the positive region is closely related to the fuzzy similarity relation, we design the second experiment using the following formula to compute the fuzzy similarity degree [38]

$$\mu_{S'_a}(x_i, x_j) = \begin{cases} 1 - 4 \times |a(x_i) - a(x_j)|, & |a(x_i) - a(x_j)| \le 0.25 \\ 0, & otherwise \end{cases} \tag{2}$$

In this experiment, we compare the reduction results of F and F1 as well as their reduction time on six datasets. The results are shown in Table 6 and Fig. 11. Table 6 shows that the reductions of the two algorithms are the same for each dataset. Fig. 11 shows that F1 is always faster than F, e.g., in the *segmentation* dataset, the time cost of F1 is 1/3 of F's, in the *pendigits* dataset, the time cost of F1 is 1/2 of F's. In fuzzy rough set model, the effectiveness of the first policy is closely related to the fuzzy similarity relation, and the second policy that optimizes the fuzzy granular structure using Algorithm 6 is more generalized.

5.4. Experiments using feature selection algorithms

We also carry out comparative experiments using feature selection algorithm [5,11], which is an important data preprocessing method outside the rough set. Similar to the reduction in the rough set, there are four basic steps in a typical

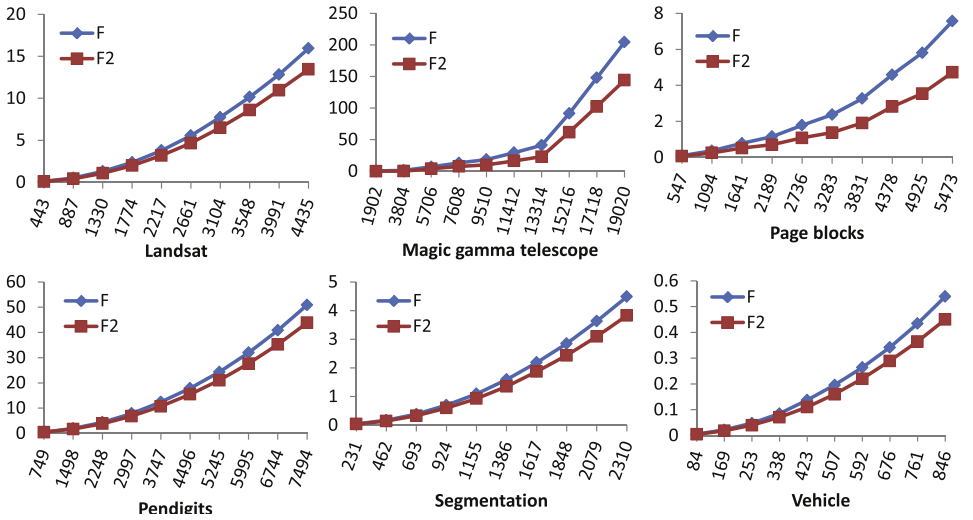

Fig. 10. The single positive region computational time in fuzzy rough set model. The X-axis represents the number of objects, and the Y-axis is the computational time (S).

Table 6
The reduction results of four algorithms using formula (2) in fuzzy rough set model.

Dataset	FAR	FAR+P1
Landsat	20,5,22,33,13,34,36,6,25, 26,12,15,21,7,35,30,23,1, 28,29,27,17,19,16,14,31,8, 11,32,10,9,2,4,3,18,24	20,5,22,33,13,34,36,6,25, 26,12,15,21,7,35,30,23,1, 28,29,27,17,19,16,14,31,8, 11,32,10,9,2,4,3,18,24
Mgt	7,9,10,4,8,6,5,3,1,2	7,9,10,4,8,6,5,3,1,2
Page blocks	6,5,2,9,4,1,10,8,3	6,5,2,9,4,1,10,8,3
Pendigits	4,8,9,15,1,16,5,3,11, 10,2,7,13,12,6,14	4,8,9,15,1,16,5,3,11, 10,2,7,13,12,6,14
Segmentation	19,12,2,14,1,16,18,4,5, 8,6,11	19,12,2,14,1,16,18,4,5, 8,6,11
Vehicle	12,10,18,15,16,1,3,13,4, 8,17,2,5,9,6	12,10,18,15,16,1,3,13,4, 8,17,2,5,9,6

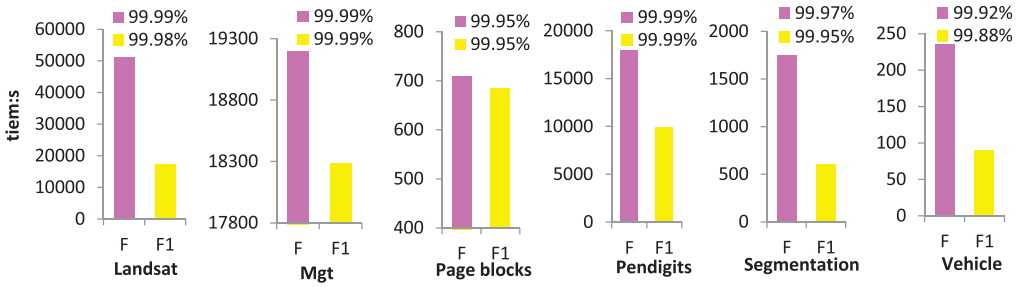

Fig. 11. The reduction computational time using formula (2) in fuzzy rough set model. The Y-axis represents the computational time (S), and the legend denotes the ratio of the computational time between the positive region and the reduction.

feature selection method, i.e. feature evaluation, search strategies, stopping criterion, and validation strategies. Many evaluation criteria are used to measure the quality of the candidate subset. Then, we choose two typical evaluation criteria to design experiments. In each group of experiments, the compared algorithms select the same features on the same dataset. Here, we no longer provide details about the feature selection results.

The first evaluation criterion is the consistency measure [5] that captures the natural objective of feature selection. We employ the forward greedy search strategy and set the inconsistency rate threshold to 5%. We use CFS to denote the generalized consistency based feature selection algorithm without any acceleration policies. We use CFS1 and CFS2 to denote the algorithms that employ the first and second acceleration policies, respectively. CFS12 denotes the algorithm with both acceleration policies. We compare the efficiency of those four algorithms on 6 datasets, i.e., *balance scale, car evaluation, chess, connect-4, mushroom,* and *nursery* in Table 2. Fig. 12 shows that CFS12 always results in a greater computational time than the other three algorithms, CFS1 clearly performs better than CFS except on the *mushroom* dataset. We find that only

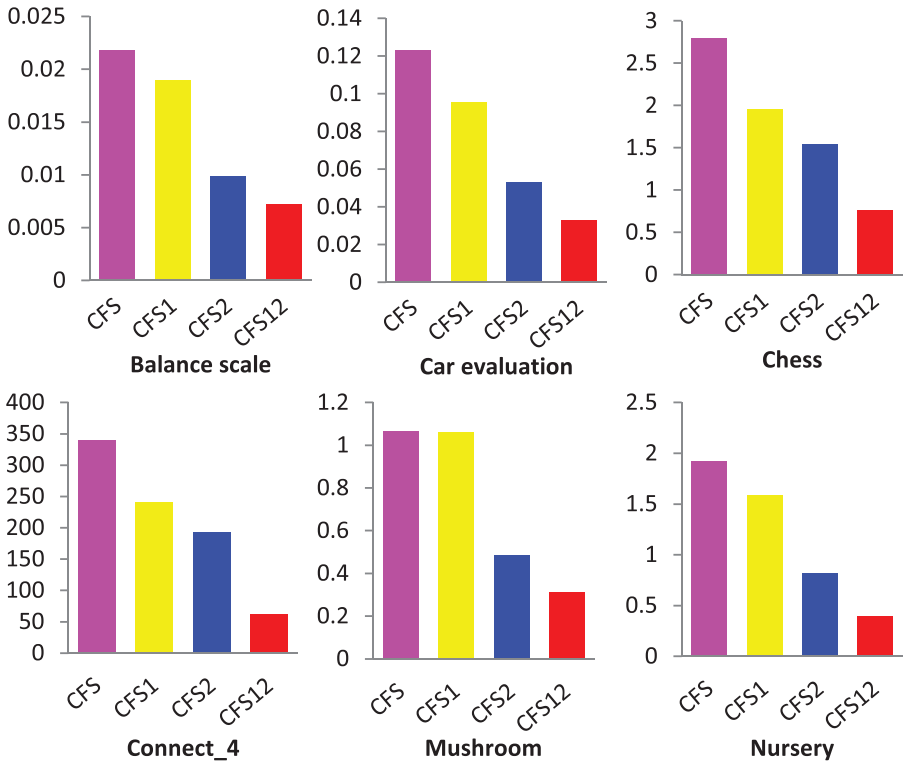

Fig. 12. The computational time of consistency based feature selection algorithms. The Y-axis is the computational time (S).

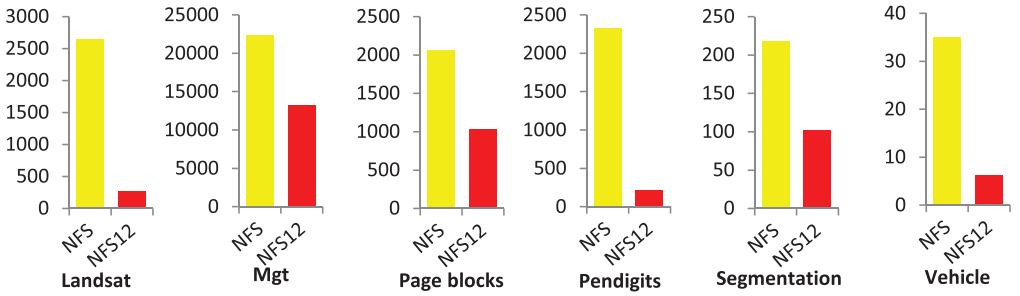

Fig. 13. The computational time of NDER based feature selection algorithms. The Y-axis is the computational time (S).

one feature is selected as the feature-selection result in the *mushroom* dataset, so the first acceleration policy does not work on this dataset. Fig. 12 shows that the first and second acceleration policies can significantly improve the efficiency of the consistency based feature selection algorithm.

The second evaluation criterion is the neighborhood decision error rate(NDER) measure [11], which minimizes the classification risk of the selected feature subset. In our experiments, NFS is the sequentially forward selection algorithm based on NDER(SFS-NDER in [11]), and NFS12 is the improved NFS algorithm with both the first and second acceleration policies. The following experiment employs six datasets, i.e., *landsat, mgt, page blocks, pendigits, segmentation,* and *vehicle*. Fig. 13 shows the computational time of the above two algorithms. The results show that NFS12 always performs better than NFS, especially on the *landsat* and *pendigits* datasets, and the time costs of NFS12 are only 1/10 of that of NFS. Thus our acceleration policies can also significantly improve the efficiency of NDER based feature selection algorithms.

6. Conclusion and future work

Attribute reduction based on rough sets is an existing method that deals with the large amounts of data. Despite the presence of many fast attribute reduction algorithms, they closely depend on the specific indiscernibility relation. In this paper, we proposed a generalized indiscernibility reduction model in which the concept of the granular structure is a quantitative measurement induced from different indiscernibility relations. On basis of the generalized model, we present a

generalized attribute reduction algorithm and a generalized positive region computing algorithm. Based on the granular structure, we quantitatively analyze the complexity of the two algorithms. We proposed acceleration policies for the two generalized algorithms, and we designed fast approaches in the three typical reduction models, i.e., the classical rough set model, neighborhood rough set model, and fuzzy rough set model. The experiments show that the application of our fast approaches to the three typical reduction models can achieve expected efficiency. With respect to future work, we will try to present more generalized indiscernibility reduction model that incorporates the fuzzy rough set models that are based on various fuzzy approximation operators. We will also try to extend generalized attribute reduction models to a parallel framework, this will be helpful to explore complex Big Data problems using granular computing theory.

Acknowledgements

This work was supported in part by the Natural Science Foundation of China under Grant U1509210 and Grant 61370173, and in part by the Natural Science Foundation of Zhejiang Province under Grant LR13F030003.

References

[1] R.B. Bhatt, M. Gopal, On fuzzy-rough sets approach to feature selection, Pattern Recognit. Lett. 26 (7) (2005) 965–975.
[2] H. Chen, T. Li, D. Ruan, J. Lin, C. Hu, A rough-set-based incremental approach for updating approximations under dynamic maintenance environments, IEEE Trans. Knowl. Data Eng. 25 (2) (2013) 274–284.
[3] X. Chen, Q. Li, Construction of rough approximations in fuzzy setting, Fuzzy Sets Syst. 158 (23) (2007) 2641–2653.
[4] A. Chouchoulas, Q. Shen, Rough set-aided keyword reduction for text categorization, Appl. Artif. Intell. 15 (9) (2001) 843–873.
[5] M. Dash, H. Liu, Consistency-based search in feature selection, Artif. Intell. 151 (1) (2003) 155–176.
[6] C. Degang, W. Changzhong, H. Qinghua, A new approach to attribute reduction of consistent and inconsistent covering decision systems with covering rough sets, Inf. Sci. 177 (17) (2007) 3500–3518.
[7] T. Deng, Y. Chen, W. Xu, Q. Dai, A novel approach to fuzzy rough sets based on a fuzzy covering, Inf. Sci. 177 (11) (2007) 2308–2326.
[8] D. Dubois, H. Prade, Rough fuzzy sets and fuzzy rough sets*, Int. J. General Syst. 17 (2–3) (1990) 191–209.
[9] M. Hall, Correlation-based feature selection for discrete and numeric class machine learning, in: Proceedings of 7th Intentional Conference on Machine Learning, Stanford University, 2000.
[10] N.S. Hoa, N.H. Son, Some efficient algorithms for rough set methods, in: Proceedings of the Conference of Information Processing and Management of Uncertainty in Knowledge-Based Systems, 1996, pp. 1451–1456.
[11] Q. Hu, W. Pedrycz, D. Yu, J. Lang, Selecting discrete and continuous features based on neighborhood decision error minimization, Syst. Man Cybern. Part B 40 (1) (2010) 137–150.
[12] Q. Hu, Z. Xie, D. Yu, Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation, Pattern Recognit. 40 (12) (2007) 3509–3521.
[13] Q. Hu, D. Yu, J. Liu, C. Wu, Neighborhood rough set based heterogeneous feature subset selection, Inf. Sci. 178 (18) (2008) 3577–3594.
[14] Q. Hu, D. Yu, Z. Xie, Information-preserving hybrid data reduction based on fuzzy-rough techniques, Pattern Recognit. Lett. 27 (5) (2006) 414–423.
[15] Q. Hu, D. Yu, Z. Xie, Neighborhood classifiers, Expert Syst. Appl. 34 (2) (2008) 866–876.
[16] Q. Hu, L. Zhang, D. Chen, W. Pedrycz, D. Yu, Gaussian kernel based fuzzy rough sets: model, uncertainty measures and applications, Int. J. Approx. Reason. 51 (4) (2010) 453–471.
[17] Q.-H. Hu, D.-R. Yu, Z.-X. Xie, Numerical attribute reduction based on neighborhood granulation and rough approximation, J. Softw. 19 (3) (2008) 640–649.
[18] X. Hu, N. Cercone, Learning in relational databases: a rough set approach, Comput. Intell. 11 (2) (1995) 323–338.
[19] R. Jensen, Q. Shen, Fuzzy–rough attribute reduction with application to web categorization, Fuzzy Sets Syst. 141 (3) (2004) 469–485.
[20] R. Jensen, Q. Shen, Semantics-preserving dimensionality reduction: rough and fuzzy-rough-based approaches, Knowl. Data Eng. IEEE Trans. 16 (12) (2004) 1457–1471.
[21] R. Jensen, Q. Shen, Fuzzy-rough data reduction with ant colony optimization, Fuzzy Sets Syst. 149 (1) (2005) 5–20.
[22] R. Jensen, Q. Shen, Fuzzy-rough sets assisted attribute selection, Fuzzy Syst. IEEE Trans. 15 (1) (2007) 73–89.
[23] R. Jensen, Q. Shen, New approaches to fuzzy-rough feature selection, Fuzzy Syst. IEEE Trans. 17 (4) (2009) 824–838.
[24] X. Jia, L. Shang, B. Zhou, Y. Yao, Generalized attribute reduct in rough set theory, Knowl. Based Syst. 91 (2016) 204–218.
[25] T.-J. Li, Y. Leung, W.-X. Zhang, Generalized fuzzy rough approximation operators based on fuzzy coverings, Int. J. Approximate Reasoning 48 (3) (2008) 836–856.
[26] H. Liu, H. Motoda, Feature Extraction, Construction and Selection: A Data Mining Perspective, Springer Science & Business Media, 1998.
[27] S.-H. Liu, Q.-J. Sheng, B. Wu, Z.-Z. Shi, F. Hu, et al., Research on efficient algorithms for rough set methods, Chin. J. Comput. 26 (5) (2003) 524–529.
[28] X. Liu, W. Pedrycz, T. Chai, M. Song, The development of fuzzy rough sets with the use of structures and algebras of axiomatic fuzzy sets, Knowl. Data Eng. IEEE Trans. 21 (3) (2009) 443–462.
[29] Y. Liu, Y. Jiang, J. Yang, Feature reduction with inconsistency, Int. J. Cognit. Inf. Natural Intell. (IJCINI) 4 (2) (2010) 77–87.
[30] Y. Liu, R. Xiong, J. Chu, Quick attribute reduction algorithm with hash, Chin. J. Comput. 32 (8) (2009) 1493–1499.
[31] Y. Liu, C. Xu, Q. Zhang, Y. Pan, Rough rule extracting from various conditions: incremental and approximate approaches for inconsistent data, Fundam. Inform. 84 (3, 4) (2008) 403–427.
[32] N. Mac Parthaláin, R. Jensen, Q. Shen, R. Zwiggelaar, Fuzzy-rough approaches for mammographic risk analysis, Intell. Data Anal. 14 (2) (2010) 225–244.
[33] Z. Meng, Z. Shi, A fast approach to attribute reduction in incomplete decision systems with tolerance relation-based rough sets, Inf. Sci. 179 (16) (2009) 2774–2793.
[34] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (5) (1982) 341–356.
[35] Z. Pawlak, Rough Sets-Theoretical Aspect of Reasoning about Data, 1991.
[36] J. Qian, D. Miao, Z. Zhang, W. Li, Hybrid approaches to attribute reduction based on indiscernibility and discernibility relation, Int. J. Approx. Reason. 52 (2) (2011) 212–230.
[37] Y. Qian, J. Liang, W. Pedrycz, C. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artif. Intell. 174 (9) (2010) 597–618.
[38] Y. Qian, Q. Wang, H. Cheng, J. Liang, C. Dang, Fuzzy-rough feature selection accelerator, Fuzzy Sets Syst. 258 (2015) 61–78.
[39] A. Skowron, Extracting laws from decision tables: a rough set approach, Comput. Intell. 11 (2) (1995) 371–388.
[40] A. Skowron, C. Rauszer, The discernibility matrices and functions in information systems, in: Intelligent Decision Support, Springer, 1992, pp. 331–362.
[41] A. Skowron, J. Stepaniuk, Tolerance approximation spaces, Fundam. Inform. 27 (2) (1996) 245–253.
[42] R. Slowinski, D. Vanderpooten, A generalized definition of rough approximations based on similarity, IEEE Trans. Knowl. Data Eng. 12 (2) (2000) 331–336.
[43] R.W. Swiniarski, A. Skowron, Rough set methods in feature selection and recognition, Pattern Recognit. Lett. 24 (6) (2003) 833–849.

[44] E.C. Tsang, D. Chen, D.S. Yeung, X.-Z. Wang, J.W. Lee, Attributes reduction using fuzzy rough sets, Fuzzy Syst. IEEE Trans. 16 (5) (2008) 1130–1141.
[45] S. Tsumoto, A new framework for incremental rule induction based on rough sets, in: 2011 IEEE International Conference on Granular Computing, 2011.
[46] F. Wang, J. Liang, Y. Qian, Attribute reduction: a dimension incremental strategy, Knowl. Based Syst. 39 (2013) 95–108.
[47] G.-Y. Wang, H. Yu, D. Yang, et al., Decision table reduction based on conditional information entropy, Chin. J. Comput. 25 (7) (2002) 759–766.
[48] X. Wang, E.C. Tsang, S. Zhao, D. Chen, D.S. Yeung, Learning fuzzy rules from fuzzy samples based on rough set technique, Inf. Sci. 177 (20) (2007) 4493–4514.
[49] W. Wei, J. Liang, Y. Qian, A comparative study of rough sets for hybrid data, Inf. Sci. 190 (2012) 1–16.
[50] W.-Z. Wu, J.-S. Mi, W.-X. Zhang, Generalized fuzzy rough sets, Inf. Sci. 151 (2003) 263–282.
[51] Y. Yao, Combination of rough and fuzzy sets based on α-level sets, in: Rough sets and Data Mining, Springer, 1997, pp. 301–321.
[52] Y. Yao, Generalized rough set models, Rough Sets in Knowledge Discovery 1, 1998a, pp. 286–318.
[53] Y. Yao, Relational interpretations of neighborhood operators and rough set approximation operators, Inf. Sci. 111 (1) (1998b) 239–259.
[54] L. Yong, H. Wenliang, J. Yunliang, Z. Zhiyong, Quick attribute reduct algorithm for neighborhood rough set model, Inf. Sci. 271 (2014) 65–81.
[55] L. Yu, H. Liu, Efficient feature selection via analysis of relevance and redundancy, J. Mach. Learn. Res. 5 (2004) 1205–1224.
[56] J. Yunliang, Y. Zhangxian, L. Yong, Quick distribution reduction algorithm in inconsistent information system, Acta Autom. Sin. 38 (3) (2012) 382–388.
[57] L.A. Zadeh, Similarity relations and fuzzy orderings, Inf. Sci. 3 (2) (1971) 177–200.
[58] J. Zhang, J.-S. Wong, Y. Pan, T. Li, A parallel matrix-based method for computing approximations in incomplete information systems, Knowl. Data Eng. IEEE Trans. 27 (2) (2015) 326–339.
[59] W.-X. Zhang, J.-S. Mi, W.-Z. Wu, Approaches to knowledge reductions in inconsistent systems, Int. J. Intell. Syst. 18 (9) (2003) 989–1000.
[60] S. Zhao, E.C. Tsang, D. Chen, The model of fuzzy variable precision rough sets, Fuzzy Syst. IEEE Trans. 17 (2) (2009) 451–467.
[61] N. Zhong, J. Dong, S. Ohsuga, Using rough sets with heuristics for feature selection, J. Intell. Inf. Syst. 16 (3) (2001) 199–214.