PAPER

On the Properties and Applications of Inconsistent Neighborhood in Neighborhood Rough Set Models

Shujiao LIAO[†,††a)], **Qingxin ZHU**[†b)], *Nonmembers*, *and* **Rui LIANG**[†c)], *Student Member*

SUMMARY Rough set theory is an important branch of data mining and granular computing, among which neighborhood rough set is presented to deal with numerical data and hybrid data. In this paper, we propose a new concept called inconsistent neighborhood, which extracts inconsistent objects from a traditional neighborhood. Firstly, a series of interesting properties are obtained for inconsistent neighborhoods. Specially, some properties generate new solutions to compute the quantities in neighborhood rough set. Then, a fast forward attribute reduction algorithm is proposed by applying the obtained properties. Experiments undertaken on twelve UCI datasets show that the proposed algorithm can get the same attribute reduction results as the existing algorithms in neighborhood rough set domain, and it runs much faster than the existing ones. This validates that employing inconsistent neighborhoods is advantageous in the applications of neighborhood rough set. The study would provide a new insight into neighborhood rough set theory.

key words: *inconsistent neighborhood, neighborhood rough set, properties, attribute reduction, fast forward algorithm, run-time*

1. Introduction

Rough set theory, as an important branch of data mining and granular computing, is an effective tool to address the uncertainty and granulation of data [1]–[3]. Up to now, except classical rough set [4], a number of extended models of rough set have been developed, such as decision-theoretic rough set [5], [6], dominance-based rough set [7], multi-granulation rough set [8], [9] and fuzzy rough set [1], [10]. Although each kind of rough set model has its own characteristic, there are some common concepts among these models, such as lower and upper approximations, positive region and boundary region. In particular, attribute reduction, also called attribute selection or feature selection, is one of key issues in the rough set domain. It refers to selecting a suitable attribute subset, also called a reduct, to reduce the data dimensionality and meanwhile to keep the ability of original decision system [11]–[13].

Most of rough set models are only applicable for nominal data, whereas numerical data and hybrid data exist widely in real applications [14]. To deal with the two kinds of data, some discretization methods were employed in data preprocessing to transform numerical attributes into nominal attributes [15], [16], but information loss may occur in the process. To address this issue, neighborhood rough set was proposed by Hu [17], [18], which has been verified to be a powerful mechanism to handle numerical data and hybrid data. In fact, before that, Lin [19] had regarded the equivalence classes in classical rough set as neighborhoods. Neighborhood rough set can be seen as a generalization of this idea, in which neighborhoods are generated by using a certain criterion (usually a specific distance function).

Neighborhoods play a crucial role in neighborhood rough set models. For an object, its neighborhoods often contain not only the objects with the same class as it but also those with different classes from it, which can be called consistent objects and inconsistent objects respectively. In this paper, we extract the inconsistent objects from neighborhoods and introduce a new concept called inconsistent neighborhood. For a given object, its inconsistent neighborhoods include only the objects whose classes differ from it. Obviously, an inconsistent neighborhood is the subset of the corresponding neighborhood. By using inconsistent neighborhoods, the consistent objects in the neighborhoods need not be considered again.

In the study, firstly the properties of inconsistent neighborhood are discussed thoroughly, and a typical example is presented to illustrate the obtained properties. The theoretical analyses reveal that the introduction of inconsistent neighborhood gives some new formulations for existing fundamental notions in neighborhood rough set, and at the same time provides some new solutions for computing the quantities in neighborhood rough set. These new solutions are usually more direct and more quick to obtain the results than the previous solutions that use neighborhoods. Then, a fast forward attribute reduction algorithm is designed through employing the properties of inconsistent neighborhood, and experiments are undertaken upon twelve datasets from the UCI (University of California - Irvine) library [20]. Experimental results indicate that the proposed algorithm can obtain the same reducts as the existing algorithms based on neighborhoods in the domain of neighborhood rough set, and it is significantly more efficient than the existing ones. Hence, to some extent using inconsistent neighborhoods is advantageous over using traditional neighborhoods in the applications. This work would offer a new view on the theory of neighborhood rough set.

The rest of the paper is organized as follows. In Sect. 2,

Manuscript received July 21, 2017.
Manuscript revised November 2, 2017.
Manuscript publicized December 20, 2017.

[†]The authors are with the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China.
[††]The author is with the School of Mathematics and Statistics, Minnan Normal University, Zhangzhou 363000, China.
a) E-mail: sjliao2011@163.com (Corresponding author)
b) E-mail: qxzhu@uestc.edu.cn (Corresponding author)
c) E-mail: harry8384@163.com

DOI: 10.1587/transinf.2017EDP7238

we mainly review the fundamental concepts and properties in traditional neighborhood rough set models. In Sect. 3, we introduce the concept of inconsistent neighborhood and discuss relevant properties thoroughly. In Sect. 4, we propose the fast forward attribute reduction algorithm by using the properties of inconsistent neighborhood. Experimental results are analyzed in Sect. 5. Finally, we conclude the paper in Sect. 6.

2. Key Concepts and Properties in Traditional Neighborhood Rough Set Models

In this section, we review and analyze the key concepts and properties in rough sets, especially neighborhood rough set. Some examples are given to illustrate these concepts and properties.

Decision system, which was formally defined in [21], is a fundamental concept in data mining and machine learning.

Definition 2.1: A *decision system* S is the 5-tuple:

$$S = (U, C, D, V = \{V_a | a \in C \cup D\}, I = \{I_a | a \in C \cup D\}),$$

where U is a finite set of objects called the universe, C is the set of condition attributes, D is the set of decision attributes with only discrete values, V_a is the set of values for each $a \in C \cup D$, and $I_a : U \to V_a$ is an information function for each $a \in C \cup D$.

In most applications, $D = \{d\}$, namely $|D| = 1$. If $|D| > 1$, we can construct $|D|$ decision systems, with each having only one decision attribute. Moreover, the decision attribute values are often called decisions for brevity.

In neighborhood rough set models, the decision system is also called neighborhood decision system, and the attribute values of numerical condition attributes are often normalized to facilitate the data processing. An example of neighborhood decision system is listed in Table 1, where $C = \{a_1, a_2, a_3, a_4\}$, a_1, a_2 are numerical attributes, and a_3, a_4 are nominal attributes. The values of numerical attributes have been normalized, while those of nominal attributes remain unchanged. The normalization approach is to employ the linear function $y = (x - min)/(max - min)$, where x is the initial value, y is the normalized value, and min and max are the minimal value and the maximal value in the attribute domain, respectively.

Table 1 An example of neighborhood decision system.

Objects	a_1	a_2	a_3	a_4	Classes
x_1	0.8235	0.9762	H	F	1
x_2	0.0392	0	H	S	1
x_3	0.7451	0.0655	A	S	1
x_4	0.9216	0.4643	H	F	2
x_5	0.5294	0.1548	A	S	2
x_6	0.0392	0.1667	H	S	2
x_7	0.8628	0.3571	R	F	3
x_8	0.1765	0.381	R	F	3
x_9	0.7451	0.3869	R	F	3

Neighborhood granule, also called neighborhood for short, plays an important role in neighborhood rough set models. We revise its original definition in [17] to obtain a new definition which is more explicit for hybrid decision systems.

Definition 2.2: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $x_i \in U$, $B \subseteq C$ and $\delta > 0$. The neighborhood of x_i with respect to attribute subset B and neighborhood radius δ is defined as:

$$\begin{aligned}\delta_B(x_i) = \{x_j \in U | (\forall a \in B_o, v(x_j, a) = v(x_i, a))\\ \wedge (\Delta_{B_u}(x_i, x_j) \leq \delta)\},\end{aligned} \tag{1}$$

where $v(x, a)$ denotes the value of object x on attribute a, $\wedge$ is "and" operator, Δ is a distance function, and B_o, B_u are the subsets of B which contain all nominal attributes and all numerical attributes in B respectively, namely $B_o \cup B_u = B$ and $B_o \cap B_u = \emptyset$.

Assuming that $x_1, x_2 \in U$ and $B_u = (a_1, a_2, \ldots, a_K)$, then a frequently-used metric, named Minkowsky distance [22], is formulated as

$$\Delta_p(x_1, x_2) = (\sum_{i=1}^{K} |v(x_1, a_i) - v(x_2, a_i)|^p)^{1/p}.$$

In this paper we use Euclidean distance Δ_2, which is

$$\Delta_2(x_1, x_2) = \sqrt{\sum_{i=1}^{K} |v(x_1, a_i) - v(x_2, a_i)|^2}. \tag{2}$$

Let $\delta = 0.15$, we compute the neighborhood $\delta_B(x)$ for any attribute subset B of the decision system shown in Table 1. Some exemplary results are given in Table 2, where B takes values listed as column headers.

Obviously, for any object, its neighborhoods change with attributes B and radius δ. Two types of monotonicity were obtained for neighborhoods in [17].

Proposition 2.3: (Type-1 monotonicity). Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B_1 \subseteq B_2 \subseteq C$. We have $\forall x \in U$, $\delta_{B_1}(x) \supseteq \delta_{B_2}(x)$.

Proposition 2.4: (Type-2 monotonicity). Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$, $\delta_1 \leq \delta_2$. We have $\forall x \in U$, $\delta_1(x) \subseteq \delta_2(x)$, where $\delta_i(x)$ denotes the neighborhood of x with respect to attributes B and radius δ_i.

Lower and upper approximations, positive region and boundary region are fundamental issues in rough set theory. They were defined in neighborhood rough set context in [17].

Definition 2.5: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, and $X_1, X_2, \ldots, X_K$ be the object subsets with decisions 1 through K. The lower and upper approximations of decision D with respect to $B \subseteq C$ are defined as

$$\underline{N_B}D = \bigcup_{i=1}^{K} \underline{N_B}(X_i), \overline{N_B}D = \bigcup_{i=1}^{K} \overline{N_B}(X_i), \tag{3}$$

Table 2 Neighborhoods of objects on some attribute subsets with $\delta = 0.15$.

x	$\{a_1\}$	$\{a_2\}$	$\{a_3\}$	$\{a_1, a_2\}$	$\{a_1, a_3\}$	$\{a_2, a_3\}$	$\{a_1, a_2, a_3\}$
x_1	$\{x_1, x_3, x_4, x_7, x_9\}$	$\{x_1\}$	$\{x_1, x_2, x_4, x_6\}$	$\{x_1\}$	$\{x_1, x_4\}$	$\{x_1\}$	$\{x_1\}$
x_2	$\{x_2, x_6, x_8\}$	$\{x_2, x_3\}$	$\{x_1, x_2, x_4, x_6\}$	$\{x_2\}$	$\{x_2, x_6\}$	$\{x_2\}$	$\{x_2\}$
x_3	$\{x_1, x_3, x_7, x_9\}$	$\{x_2, x_3, x_5, x_6\}$	$\{x_3, x_5\}$	$\{x_3\}$	$\{x_3\}$	$\{x_3, x_5\}$	$\{x_3\}$
x_4	$\{x_1, x_4, x_7\}$	$\{x_4, x_7, x_8, x_9\}$	$\{x_1, x_2, x_4, x_6\}$	$\{x_4, x_7\}$	$\{x_1, x_4\}$	$\{x_4\}$	$\{x_4\}$
x_5	$\{x_5\}$	$\{x_3, x_5, x_6\}$	$\{x_3, x_5\}$	$\{x_5\}$	$\{x_5\}$	$\{x_3, x_5\}$	$\{x_5\}$
x_6	$\{x_2, x_6, x_8\}$	$\{x_3, x_5, x_6\}$	$\{x_1, x_2, x_4, x_6\}$	$\{x_6\}$	$\{x_2, x_6\}$	$\{x_6\}$	$\{x_6\}$
x_7	$\{x_1, x_3, x_4, x_7, x_9\}$	$\{x_4, x_7, x_8, x_9\}$	$\{x_7, x_8, x_9\}$	$\{x_7, x_9\}$	$\{x_7, x_9\}$	$\{x_7, x_8, x_9\}$	$\{x_7, x_9\}$
x_8	$\{x_2, x_6, x_8\}$	$\{x_4, x_7, x_8, x_9\}$	$\{x_7, x_8, x_9\}$	$\{x_8\}$	$\{x_8\}$	$\{x_7, x_8, x_9\}$	$\{x_8\}$
x_9	$\{x_1, x_3, x_7, x_9\}$	$\{x_4, x_7, x_8, x_9\}$	$\{x_7, x_8, x_9\}$	$\{x_7, x_9\}$	$\{x_7, x_9\}$	$\{x_7, x_8, x_9\}$	$\{x_7, x_9\}$

where

$$\begin{aligned} \underline{N_B}(X) &= \{x \in U | \delta_B(x) \subseteq X\}, \\ \overline{N_B}(X) &= \{x \in U | \delta_B(x) \cap X \neq \emptyset\} \end{aligned} \tag{4}$$

are the lower and upper approximations of object subset X. The boundary region of decision D with respect to attributes B is defined as

$$BN_B(D) = \overline{N_B}D - \underline{N_B}D.$$

The lower approximation $\underline{N_B}D$ is also called positive region and is denoted by $POS_B(D)$. If not specified, the lower and upper approximations refer to those of object subsets in the following. The relations between above concepts were given in [17], which are (1) $\overline{N_B}D = U$; (2) $POS_B(D) \bigcap BN_B(D) = \emptyset$; (3) $POS_B(D) \bigcup BN_B(D) = U$. From the relations, it is known that

$$BN_B(D) = U - POS_B(D). \tag{5}$$

Reduct, as an important concept in rough sets, is an attribute subset that has the same approximating power as the whole set of attributes. The definition of decision-relative reduct was given in [23].

Definition 2.6: Let $S = (U, C, D, V, I)$ be a decision system. Any $B \subseteq C$ is a decision-relative reduct if:
(1) $POS_B(D) = POS_C(D)$;
(2) $\forall a \in B, POS_{B-\{a\}}(D) \subset POS_B(D)$.

3. Inconsistent Neighborhood and Relevant Properties

In this section, we introduce the concept of inconsistent neighborhood, and discuss the relevant properties thoroughly. A representative example is given to illustrate the obtained properties.

We start from introducing consistent objects and inconsistent objects in the neighborhoods.

Definition 3.1: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$ and $x \in U$. For any $y \in \delta_B(x)$, if $D(y) = D(x)$, y is called a consistent object in $\delta_B(x)$; otherwise, y is called an inconsistent object in $\delta_B(x)$.

For example, it is known from Table 2 that in $\delta_{a_1}(x_1)$, x_3 is a consistent object, while x_4, x_7 and x_9 are all inconsistent objects. Inconsistent neighborhood, which refers to the set of inconsistent objects in a neighborhood, is defined as follows.

Definition 3.2: Let $S = (U, C, D, V, I)$ be a neighborhood decision system. Given $x_i \in U, B \subseteq C$ and $\delta > 0$, the inconsistent neighborhood of x_i with respect to attribute subset B and neighborhood radius δ is defined as

$$in_B(x_i) = \{x_j \in U | x_j \in \delta_B(x_i), D(x_j) \neq D(x_i)\}. \tag{6}$$

Naturally, according to Definition 2.2, Eq. (6) is equivalent to

$$\begin{aligned} in_B(x_i) = \{x_j \in U | &(\forall a \in B_o, v(x_j, a) = v(x_i, a)) \\ &\wedge (\Delta_{B_u}(x_i, x_j) \leq \delta), D(x_j) \neq D(x_i)\}, \end{aligned} \tag{7}$$

where B_o and B_u have been introduced in Definition 2.2. Hence, the inconsistent neighborhoods can be calculated on basis of neighborhoods, or by using Eq. (7) directly.

In the following, we will explore the properties of inconsistent neighborhood. It is notable that, in [24] the set of inconsistent objects, namely inconsistent neighborhood in this paper, has been used to find a test-cost-sensitive reduct in error-range-based covering rough set model. Since the data model has changed in this paper, the properties of inconsistent neighborhood need to be restudied. Moreover, as will be shown below, the properties are discussed more thoroughly in our work, and some of them are relatively different from those in [24].

Proposition 3.3: Let $S = (U, C, D, V, I)$ be a neighborhood decision system. Given any $x, x_i, x_j \in U$ and $B \subseteq C$, we have
(1) $in_\emptyset(x) = \{y \in U | D(y) \neq D(x)\}$;
(2) $in_B(x) \subset \delta_B(x)$;
(3) $x_j \in in_B(x_i) \Leftrightarrow x_i \in in_B(x_j)$;
(4) $x \in POS_B(D) \Leftrightarrow in_B(x) = \emptyset$.

Proof 3.4: (1)–(3) can be known immediately from Definition 3.2 and Eq. (7).
(4) Let $X_1, X_2, \ldots, X_K$ be the object subsets with decisions 1 through K. $x \in POS_B(D) \Leftrightarrow \exists X_i (1 \leq i \leq K), \delta_B(x) \subseteq X_i \Leftrightarrow \forall y \in \delta_B(x), D(y) = D(x) \Leftrightarrow in_B(x) = \emptyset$.

Note that, according to the essence of lower and upper approximations of object subsets, we can rewrite Eq. (4) as follows:

$$\begin{aligned} \underline{N_B}(X) &= \{x \in X | \delta_B(x) \subseteq X\}, \\ \overline{N_B}(X) &= X \cup \{x \notin X | \delta_B(x) \cap X \neq \emptyset\}. \end{aligned} \tag{8}$$

Compared with Eq. (4), Eq. (8) is more explicit, and the computational efficiency of lower and upper approximations can be improved by using it. Further, according to Definition 3.2, Eq. (8) can be rewritten in a new form by using inconsistent neighborhoods.

Proposition 3.5: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$. For any $X \subseteq U$, we have

$$\begin{aligned} &\underline{N_B}(X) = \{x \in X | in_B(x) = \emptyset\}, \\ &\overline{N_B}(X) = X \cup \{x \notin X | in_B(x) \cap X \neq \emptyset\}. \end{aligned} \tag{9}$$

Moreover, based on Proposition 3.3 and Eq. (5), we can obtain the following formulations for the positive region and the boundary region.

Proposition 3.6: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$. We have

$$\begin{aligned} POS_B(D) &= \{x \in U | in_B(x) = \emptyset\}, \\ BN_B(D) &= \{x \in U | in_B(x) \neq \emptyset\}. \end{aligned} \tag{10}$$

According to Definition 2.6 and Proposition 3.3, we obtain the following proposition, which can be used as an alternative definition of reduct.

Proposition 3.7: Let $S = (U, C, D, V, I)$ be a neighborhood decision system. Any $B \subseteq C$ is a decision-relative reduct if:
(1) $\forall x \in POS_C(D)$, $in_B(x) = \emptyset$;
(2) $\forall a \in B$, $\exists x \in POS_C(D)$, s.t. $in_{B-\{a\}}(x) \neq \emptyset$.

In fact, $POS_C(D) = U$ in most cases, then we have

Proposition 3.8: Let $S = (U, C, D, V, I)$ be a neighborhood decision system. Any $B \subseteq C$ is a decision-relative reduct if:
(1) $\forall x \in U$, $in_B(x) = \emptyset$;
(2) $\forall a \in B$, $\exists x \in U$, s.t. $in_{B-\{a\}}(x) \neq \emptyset$.

In general, Propositions 3.5–3.8 give new formulations for the lower and upper approximations, positive region, boundary region and reduct through employing inconsistent neighborhoods.

Interestingly, according to Proposition 3.5, we find that there are close relations between the lower approximations and the upper approximations among different classes. The relations are displayed in the following two propositions.

Proposition 3.9: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$, and let $X_1, X_2, \ldots, X_K$ be the object subsets with decisions 1 through K. We have

$$\underline{N_B}(X_k) = U - \bigcup_{i=1, i \neq k}^{K} \overline{N_B}(X_i). \tag{11}$$

Proof 3.10: We can get

$$\begin{aligned} &\bigcup_{i=1,i\neq k}^{K} \overline{N_B}(X_i) \\ &= \bigcup_{i=1,i\neq k}^{K} (X_i \cup \{x \notin X_i | in_B(x) \cap X_i \neq \emptyset\}) \\ &= (\bigcup_{i=1,i\neq k}^{K} X_i) \bigcup (\bigcup_{i=1,i\neq k}^{K} \{x \notin X_i | in_B(x) \cap X_i \neq \emptyset\}) \\ &= (\bigcup_{i=1,i\neq k}^{K} X_i) \bigcup (\bigcup_{i=1,i\neq k}^{K} \{x \in X_k | in_B(x) \cap X_i \neq \emptyset\}) \\ &= (\bigcup_{i=1,i\neq k}^{K} X_i) \bigcup \{x \in X_k | in_B(x) \bigcap (\bigcup_{i=1,i\neq k}^{K} X_i) \neq \emptyset\} \\ &= (\bigcup_{i=1,i\neq k}^{K} X_i) \bigcup \{x \in X_k | in_B(x) \neq \emptyset\}. \end{aligned}$$

Since $\{X_1, X_2, \ldots, X_K\}$ is a partition of U, and $\underline{N_B}(X_k) = \{x \in X_k | in_B(x) = \emptyset\}$, we have

$$\begin{aligned} &\underline{N_B}(X_k) \bigcup (\bigcup_{i=1,i\neq k}^{K} \overline{N_B}(X_i)) = U, \\ &\underline{N_B}(X_k) \bigcap (\bigcup_{i=1,i\neq k}^{K} \overline{N_B}(X_i)) = \emptyset. \end{aligned}$$

So $\underline{N_B}(X_k) = U - \bigcup_{i=1,i\neq k}^{K} \overline{N_B}(X_i)$.

Proposition 3.11: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$, and let $X_1, X_2, \ldots, X_K$ be the object subsets with decisions 1 through K. We have

$$\overline{N_B}(X_k) \subseteq U - \bigcup_{i=1,i\neq k}^{K} \underline{N_B}(X_i). \tag{12}$$

Proof 3.12: Since $\overline{N_B}(X_k) = X_k \cup \{x \notin X_k | in_B(x) \cap X_k \neq \emptyset\}$,

$$\begin{aligned} &U - \bigcup_{i=1,i\neq k}^{K} \underline{N_B}(X_i) \\ &= U - \bigcup_{i=1,i\neq k}^{K} \{x \in X_i | in_B(x) = \emptyset\} \\ &= X_k \bigcup (\bigcup_{i=1,i\neq k}^{K} \{x \in X_i | in_B(x) \neq \emptyset\}) \\ &= X_k \cup \{x \notin X_k | in_B(x) \neq \emptyset\}, \end{aligned}$$

and $\{x \notin X_k | in_B(x) \cap X_k \neq \emptyset\} \subseteq \{x \notin X_k | in_B(x) \neq \emptyset\}$, we have $\overline{N_B}(X_k) \subseteq U - \bigcup_{i=1,i\neq k}^{K} \underline{N_B}(X_i)$.

Similarly with neighborhoods, there are two types of monotonicity for inconsistent neighborhoods according to Propositions 2.3–2.4 and Definition 3.2.

Proposition 3.13: (Type-1 monotonicity). Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B_1 \subseteq B_2 \subseteq C$.

We have $\forall x \in U$, $in_{B_1}(x) \supseteq in_{B_2}(x)$.

Proposition 3.14: (Type-2 monotonicity). Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$, $\delta_1 \leq \delta_2$. We have $\forall x \in U$, $in_1(x) \subseteq in_2(x)$, where $in_i(x)$ denotes the inconsistent neighborhood of x with respect to attributes B and radius δ_i.

Furthermore, we can obtain the following two corollaries.

Corollary 3.15: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, $B \subseteq C$. Assuming that $B_i \subseteq B$ ($1 \leq i \leq L$, L is a finite positive integer), then for any $x \in U$, we have

$$in_B(x) \subseteq \bigcap_{i=1}^{L} in_{B_i}(x) \subseteq \bigcup_{i=1}^{L} in_{B_i}(x).$$

Proof 3.16: Based on Proposition 3.13, for any $x \in U$ and any B_i, we have $in_B(x) \subseteq in_{B_i}(x)$, so $in_B(x) \subseteq \bigcap_{i=1}^{L} in_{B_i}(x) \subseteq \bigcup_{i=1}^{L} in_{B_i}(x)$.

Corollary 3.17: Let $S = (U, C, D, V, I)$ be a neighborhood decision system, and $B = \{a_1, a_2, \cdots, a_n\} \subseteq C$. Assuming that $\{B_i\}_{1 \leq i \leq L}(2 \leq L \leq n)$ are disjoint subsets of B which satisfy $\bigcup_{i=1}^{L} B_i = B$, then for any $x \in U$, we have

$$in_B(x) \subseteq \bigcap_{i=1}^{L} in_{B_i}(x),$$

where "=" holds when at most one B_i contains numerical attributes.

Proof 3.18: Based on Corollary 3.15, we can obtain immediately $in_B(x) \subseteq \bigcap_{i=1}^{L} in_{B_i}(x), \forall x \in U$. Now under three cases as follows, we prove that $in_B(x) = \bigcap_{i=1}^{L} in_{B_i}(x), \forall x \in U$ holds when at most one B_i contains numerical attributes:
(1) None of $B_i (i = 1, 2, \cdots, L)$ contains numerical attributes, namely all attributes in B are nominal. According to Eq. (7), in this case the comparisons of attribute values are independent between different attributes in both B and each B_i, thus we can obtain $in_B(x) = \bigcap_{i=1}^{L} in_{B_i}(x)$, $\forall x \in U$.
(2) Only one B_i contains numerical attributes. Similarly with (1), the comparisons of attribute values are independent between different nominal attributes in both B and each nominal B_i. As for numerical attributes, their attribute values are used together to compute Euclidean distance based on Eq. (2) in both B and its sole numerical subset, thus we can also obtain $in_B(x) = \bigcap_{i=1}^{L} in_{B_i}(x)$, $\forall x \in U$.
(3) More than one B_i contains numerical attributes. Without loss of generality, consider the simplest case where $B_1 = \{a_1, a_2, \cdots, a_j\}$, $B_2 = \{a_{j+1}, a_{j+2}, \cdots, a_n\}$, and a_i, a_k ($1 \leq i \leq j$, $j + 1 \leq k \leq n$) are numerical attributes, namely B is composed of two disjoint subsets, each with exactly one numerical attribute. According to Eqs. (2) and (7), $\forall y \in in_{B_1}(x) \cap in_{B_2}(x)$ means $|v(y, a_i) - v(x, a_i)| \leq \delta$ and $|v(y, a_k) - v(x, a_k)| \leq \delta$, while $\forall y \in in_B(x)$ means $\sqrt{|v(y, a_i) - v(x, a_i)|^2 + |v(y, a_k) - v(x, a_k)|^2} \leq \delta$. Since $(|v(y, a_i) - v(x, a_i)| \leq \delta) \bigwedge (|v(y, a_k) - v(x, a_k)| \leq \delta) \nRightarrow \sqrt{|v(y, a_i) - v(x, a_i)|^2 + |v(y, a_k) - v(x, a_k)|^2} \leq \delta$, $in_B(x) \supseteq in_{B_1}(x) \cap in_{B_2}(x), \forall x \in U$ may not hold in this case. Similarly, we can deduce that $in_B(x) \supseteq \bigcap_{i=1}^{L} in_{B_i}(x), \forall x \in U$ may not hold for other cases where more than one B_i contains numerical attributes. Hence, $in_B(x) = \bigcap_{i=1}^{L} in_{B_i}(x), \forall x \in U$ may not hold when more than one B_i contains numerical attributes.
To sum up, $in_B(x) = \bigcap_{i=1}^{L} in_{B_i}(x), \forall x \in U$ holds when at most one B_i contains numerical attributes.

As mentioned earlier, compared with the existing work in [24], the properties of inconsistent neighborhood are explored more thoroughly in our work, and a series of new properties are obtained. Moreover, since the data environment has changed, the obtained properties may be greatly different between the existing work and our work. For example, under the condition of Corollary 3.17, $in_B(x) = \bigcap_{i=1}^{L} in_{B_i}(x), \forall x \in U$ holds when at most one B_i contains numerical attributes in our work, but it always holds in the existing work.

We give the following example to illustrate the concepts and properties discussed above.

Example 3.19: A neighborhood decision system is indicated by Table 1, from which it is known that $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$, $C = \{a_1, a_2, a_3, a_4\}$ and $U/D = \{\{x_1, x_2, x_3\}, \{x_4, x_5, x_6\}, \{x_7, x_8, x_9\}\}$. In other words, the objects in the universe are grouped into three subsets according to the decision attribute. Let $X_1 = \{x_1, x_2, x_3\}$, $X_2 = \{x_4, x_5, x_6\}$ and $X_3 = \{x_7, x_8, x_9\}$, and let $\delta = 0.15$.

We can obtain the inconsistent neighborhood $in_B(x)$ for any attribute subset B. Some exemplary results are shown in Table 3, where B takes values listed as column headers. Combined Table 3 and Proposition 3.8, it is known that $\{a_1, a_2, a_3\}$ is a reduct for the decision system.

According to Proposition 3.6, we can obtain the positive regions $POS_B(D)$ and the boundary regions $BN_B(D)$ on the exemplary attribute subsets by using the inconsistent neighborhoods shown in Table 3. Some results are listed in Table 4.

The lower and upper approximations of the three object subsets X_1, X_2, X_3 can be computed by combining Table 3 with Proposition 3.5. Some results are given in Table 5.

Finally, we give some examples for Corollary 3.17. It is known from Table 1 that, a_1, a_2 are numerical attributes while a_3 is a nominal attribute. And it can be found from Table 3 that, $\forall x_i \in U$, $in_{\{a_1,a_3\}}(x_i) = in_{\{a_1\}}(x_i) \cap in_{\{a_3\}}(x_i)$, $in_{\{a_2,a_3\}}(x_i) = in_{\{a_2\}}(x_i) \cap in_{\{a_3\}}(x_i)$, while $in_{\{a_1,a_2\}}(x_7) \subset in_{\{a_1\}}(x_7) \cap in_{\{a_2\}}(x_7)$. Corollary 3.17 can be validated on these examples.

The properties of inconsistent neighborhood can be verified from the results of Example 3.19 (Except those mentioned in Example 3.19, the verification of other properties is omitted to save the space). More importantly, through comparing Table 3 with Table 2, it is found that the inconsistent neighborhoods are usually much narrower than the corresponding neighborhoods, so the subsequent computations using inconsistent neighborhoods are often faster than

Table 3 Inconsistent neighborhoods of objects on some attribute subsets with $\delta = 0.15$.

x	$\{a_1\}$	$\{a_2\}$	$\{a_3\}$	$\{a_1, a_2\}$	$\{a_1, a_3\}$	$\{a_2, a_3\}$	$\{a_1, a_2, a_3\}$
x_1	$\{x_4, x_7, x_9\}$	$\emptyset$	$\{x_4, x_6\}$	$\emptyset$	$\{x_4\}$	$\emptyset$	$\emptyset$
x_2	$\{x_6, x_8\}$	$\emptyset$	$\{x_4, x_6\}$	$\emptyset$	$\{x_6\}$	$\emptyset$	$\emptyset$
x_3	$\{x_7, x_9\}$	$\{x_5, x_6\}$	$\{x_5\}$	$\emptyset$	$\emptyset$	$\{x_5\}$	$\emptyset$
x_4	$\{x_1, x_7\}$	$\{x_7, x_8, x_9\}$	$\{x_1, x_2\}$	$\{x_7\}$	$\{x_1\}$	$\emptyset$	$\emptyset$
x_5	$\emptyset$	$\{x_3\}$	$\{x_3\}$	$\emptyset$	$\emptyset$	$\{x_3\}$	$\emptyset$
x_6	$\{x_2, x_8\}$	$\{x_3\}$	$\{x_1, x_2\}$	$\emptyset$	$\{x_2\}$	$\emptyset$	$\emptyset$
x_7	$\{x_1, x_3, x_4\}$	$\{x_4\}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
x_8	$\{x_2, x_6\}$	$\{x_4\}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
x_9	$\{x_1, x_3\}$	$\{x_4\}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Table 4 Positive regions and boundary regions of decision D on some attribute subsets.

	$\{a_1\}$	$\{a_2\}$	$\{a_3\}$	$\{a_1, a_2\}$	$\{a_1, a_2, a_3\}$
$POS_B(D)$	$\{x_5\}$	$\{x_1, x_2\}$	$\{x_7, x_8, x_9\}$	$\{x_1, x_2, x_3, x_5, x_6, x_7, x_8, x_9\}$	U
$BN_B(D)$	$\{x_1, x_2, x_3, x_4, x_6, x_7, x_8, x_9\}$	$\{x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$	$\{x_4\}$	$\emptyset$

Table 5 Lower and upper approximations of three object subsets on some attribute subsets.

	X	$\{a_1\}$	$\{a_2\}$	$\{a_3\}$	$\{a_1, a_2\}$	$\{a_1, a_2, a_3\}$
	X_1	$\emptyset$	$\{x_1, x_2\}$	$\emptyset$	$\{x_1, x_2, x_3\}$	$\{x_1, x_2, x_3\}$
$\underline{N_B}(X)$	X_2	$\{x_5\}$	$\emptyset$	$\emptyset$	$\{x_5, x_6\}$	$\{x_4, x_5, x_6\}$
	X_3	$\emptyset$	$\emptyset$	$\{x_7, x_8, x_9\}$	$\{x_7, x_8, x_9\}$	$\{x_7, x_8, x_9\}$
	X_1	$\{x_1, x_2, x_3, x_4, x_6, x_7, x_8, x_9\}$	$\{x_1, x_2, x_3, x_5, x_6\}$	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$	$\{x_1, x_2, x_3\}$	$\{x_1, x_2, x_3\}$
$\overline{N_B}(X)$	X_2	$\{x_1, x_2, x_4, x_5, x_6, x_7, x_8\}$	$\{x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$	$\{x_4, x_5, x_6\}$	$\{x_4, x_5, x_6\}$
	X_3	$\{x_1, x_2, x_3, x_4, x_6, x_7, x_8, x_9\}$	$\{x_4, x_7, x_8, x_9\}$	$\{x_7, x_8, x_9\}$	$\{x_4, x_7, x_8, x_9\}$	$\{x_7, x_8, x_9\}$

those using neighborhoods. Combined the obtained properties with Example 3.19, it can be known that some new and efficient solutions are provided for computing the quantities (i.e., the reduct, positive region, boundary region, lower and upper approximations) in neighborhood rough set models. The new solutions are summarized as follows:

(1) In previous methods, reducts cannot be known until positive regions or related values such as dependency degrees have been computed [17]. Now the reducts can be captured according to the situation of inconsistent neighborhoods directly, which will accelerate the process of attribute reduction.

(2) In existing work, positive regions and boundary regions cannot be obtained until the lower and upper approximations of object subsets have been calculated [17]. Now they can be gained immediately by using the inconsistent neighborhoods.

(3) In previous methods, lower and upper approximations of object subsets are computed by using traditional neighborhoods according to Eq. (4). The new computation method, which uses inconsistent neighborhoods according to Eq. (9), is usually more efficient than before.

In summary, the introduction of inconsistent neighborhood gives some new formulations and some efficient solutions for the theory of neighborhood rough set.

4. Algorithm

To evaluate the effectiveness of using inconsistent neighborhoods in neighborhood rough set domain, we design a fast forward attribute reduction algorithm by employing the properties of inconsistent neighborhood in this section. The algorithm framework is shown in Algorithm 1.

Algorithm 1 A fast forward attribute reduction algorithm

Input: The neighborhood decision system $S = (U, C, D, V, I)$ and the neighborhood radius δ.
Output: A reduct R.

$R = \emptyset$; //R is the set of selected attributes
$S = U$; //S is the set of objects out of the positive region
Compute $in_\emptyset(x)$ for any $x \in U$;
while $(S \neq \emptyset)$ **do**
  for (each $a_i \in C - R$) **do**
    $IPR_i = \emptyset$; //IPR_i is the increment of positive region induced by a_i
    for (each $x_j \in S$) **do**
      $in_{R\cup\{a_i\}}(x_j) = \emptyset$;
      for (each $x_k \in in_R(x_j)$) **do**
        if $(((a_i$ is nominal$) \bigwedge (v(x_k, a_i) == v(x_j, a_i))) \bigvee ((a_i$ is numerical$) \bigwedge (\Delta_{R_u\cup\{a_i\}}(x_k, x_j) \leq \delta)))$ **then**
          $in_{R\cup\{a_i\}}(x_j) = in_{R\cup\{a_i\}}(x_j) \bigcup \{x_k\}$; //$R_u$ is the subset of R containing all numerical attributes of R
        end if
      end for
      if $(in_{R\cup\{a_i\}}(x_j) = \emptyset)$ **then**
        $IPR_i = IPR_i \cup \{x_j\}$;
      end if
    end for
  end for
  Find a_l such that $|IPR_l| = max_i|IPR_i|$; //$|IPR_i|$ is used as the significance of attribute a_i
  if $(|IPR_l| > 0)$ **then**
    $R = R \cup \{a_l\}$;
    $S = S - IPR_l$;
  else
    break;
  end if
end while
return R;

In the attribute reduction algorithm, we first compute the initial inconsistent neighborhood $in_{\emptyset}(x)$ for any object x in the universe. Then, the attributes are added into the reduct R one by one according to their significances until none of the significances is more than zero or no object lies outside the positive region. For each unselected attribute $a_i \in C - R$, its significance is measured with $|IPR_i|$, namely the size of the incremental positive region induced by a_i, and is computed through using the inconsistent neighborhoods.

We mainly use three techniques to accelerate the reduction process in Algorithm 1. The first two ones are shown in lines 9–13. Firstly, since $in_{R\cup\{a_i\}}(x) \subseteq in_R(x)$ according to Proposition 3.13, we only need to judge whether the objects in $in_R(x)$, instead of all objects in U, belong to $in_{R\cup\{a_i\}}(x)$. Secondly, by using the obtained initial inconsistent neighborhoods, in the while-loop the computation of inconsistent neighborhoods needs not use the decision attributes. Finally, by using $S = S - IPR_l$ in line 22, as the attribute reduction goes on, the objects out of the positive region get fewer and fewer, and at the same time their inconsistent neighborhoods get smaller and smaller until be equal to $\emptyset$. In general, the computation will be reduced significantly at the sequential rounds of the while-loop, and the reduction procedure will be sped up greatly.

Now the time complexity of the algorithm is analyzed, in which the computation of inconsistent neighborhoods is crucial. By using sorting technique, the time complexity is $O(n)$ for computing the initial inconsistent neighborhoods, where n is the number of objects. As for the computation of inconsistent neighborhoods at each round of the while-loop, since it needs not use the decision attributes, its time complexity is equal to that for computing traditional neighborhoods, which is $O(nlogn)$ [17]. Given a decision system with N attributes, n objects and m classes, then the initial inconsistent neighborhoods averagely contain $n \times \frac{m-1}{m}$ objects. Assuming that there are k attributes included in the reduct, and selecting an attribute averagely leads to n/k objects added into the positive region, then the total computational time of the algorithm is
$(N \times nlogn + (N-1) \times nlogn \times \frac{k-1}{k} + \cdots + (N-k) \times nlogn \times \frac{1}{k}) \times \frac{m-1}{m} < \frac{(m-1)Nnlogn}{mk}(k+k-1+\cdots+1) = \frac{(m-1)(k+1)}{2m}Nnlogn$.
In fact, by using the three accelerating techniques, the run-time of the algorithm is often much less than that in the above equation.

5. Experiments

In this section, we test the performance of the proposed fast forward attribute reduction algorithm by comparing it with the state-of-art attribute reduction algorithms in neighborhood rough set domain. It is worth noticing that, Hu et al. have proposed a naive forward attribute reduction algorithm (Algorithm 1 in [17]) and a fast forward attribute reduction algorithm (Algorithm 2 in [17]) on the basis of traditional neighborhoods, and have verified that the two algorithms can obtain the same reducts while the latter runs much faster than the former. Therefore, we only need to compare our algorithm with the existing fast algorithm. Twelve UCI datasets are used in the experiments, among which five datasets are hybrid and other seven ones are numerical. The basic information of these datasets is listed in Table 6. Before the experiments, the values of numerical attributes are normalized into $[0, 1]$, and those of nominal attributes remain unchanged.

Table 6 Dataset information.

Name	Domain	Objects	Nominal	Numerical	Classes
Credit	finance	690	9	6	2
Cylinder	physics	430	16	20	2
German	finance	1000	13	7	2
Heart	clinic	303	8	5	5
Hypothyroid	clinic	3163	18	6	2
Diabetes	clinic	768	0	8	2
Image	graphics	210	0	19	7
Ionosphere	physics	351	0	34	2
Sonar	physics	208	0	60	2
Wdbc	clinic	569	0	30	2
Wine	agriculture	178	0	13	3
Wpbc	clinic	198	0	33	2

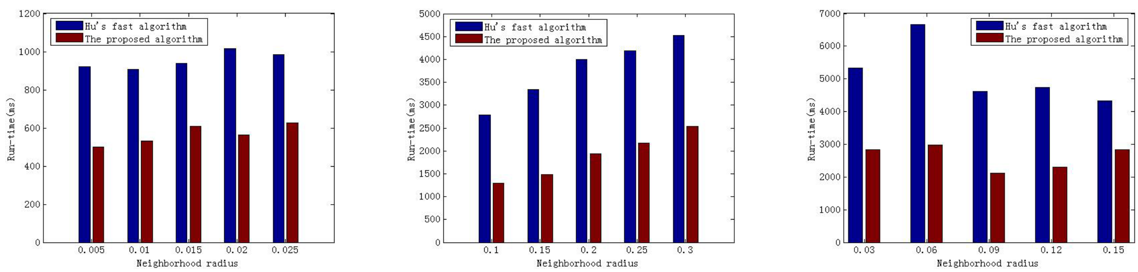
For each dataset, we run the proposed reduction algorithm and Hu's fast reduction algorithm by taking values for the neighborhood radius δ from 0.005 to 1 with step-size 0.005. As shown in Table 7, we list a group of exemplary attribute reduction results for each dataset. Interestingly, we find from the experiments that, for each dataset and each δ value, the obtained reducts are identical to each other between the two algorithms. It is notable that, as pointed out in [17], for CART and SVM classifiers, the classification accuracies of the data reduced with Hu's attribute reduction algorithms are not less than or even more than those of the raw data within a certian range of δ value. Hence, the two classifiers can get the same good classification performance on the data reduced by using our attribute reduction algorithm.

Then, we compare the run-time between the proposed reduction algorithm and Hu's fast reduction algorithm. Some representative results are shown in Figs. 1–4, in which the unit of run-time is millisecond (ms). The reason why the values of the neighborhood radius δ are not the same for all datasets is that (the reason is found from the experiments), for most numerical datasets $[0.1, 0.3]$ is a candidate interval for δ in terms of good classification performance, while for hybrid datasets the candidate intervals are often smaller. It is immediately known from the figures that, the proposed algorithm runs much more quickly than Hu's fast algorithm on each dataset. Naturally, the algorithm is also much more efficient than Hu's naive reduction algorithm.

In general, the proposed fast forward attribute reduction algorithm can obtain the same reducts as the existing attribute reduction algorithms in neighborhood rough set models, while it runs much faster than the existing algorithms. This verifies that using inconsistent neighborhoods is advantageous over using traditional neighborhoods in the applications of neighborhood rough set theory.

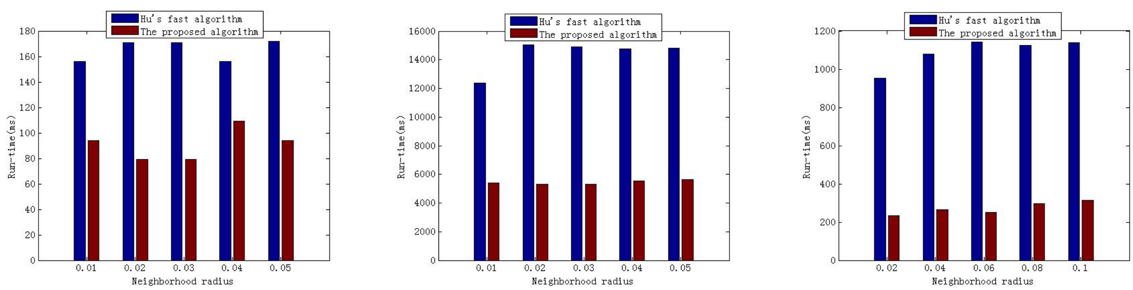
Table 7 Some exemplary reducts obtained by two attribute reduction algorithms.

Dataset	δ	Hu's fast algorithm	The proposed algorithm
Credit	0.015	2,3,8,11,14,15	2,3,8,11,14,15
Cylinder	0.2	3,20,21,22,24,25,26,27,28,31,33,36	3,20,21,22,24,25,26,27,28,31,33,36
German	0.09	2,5,8,11,13,16,18	2,5,8,11,13,16,18
Heart	0.03	1,4,5,8,10	1,4,5,8,10
Hypothyroid	0.03	1,15,17,19,21	1,15,17,19,21
Diabetes	0.06	1,2,3,4,5,6,8	1,2,3,4,5,6,8
Image	0.2	1,2,3,5,7,11,13,14,15,17,18	1,2,3,5,7,11,13,14,15,17,18
Ionosphere	0.2	1,3,4,5,7,8,12,14,19,25,30,34	1,3,4,5,7,8,12,14,19,25,30,34
Sonar	0.2	1,10,17,21,23,28,35,58	1,10,17,21,23,28,35,58
Wdbc	0.2	1,2,5,6,7,8,9,12,16,18,19,22,23,25,27,28,29,30	1,2,5,6,7,8,9,12,16,18,19,22,23,25,27,28,29,30
Wine	0.2	1,2,5,7,8,10,13	1,2,5,7,8,10,13
Wpbc	0.2	1,2,3,6,10,11,12,13,29,32,33	1,2,3,6,10,11,12,13,29,32,33



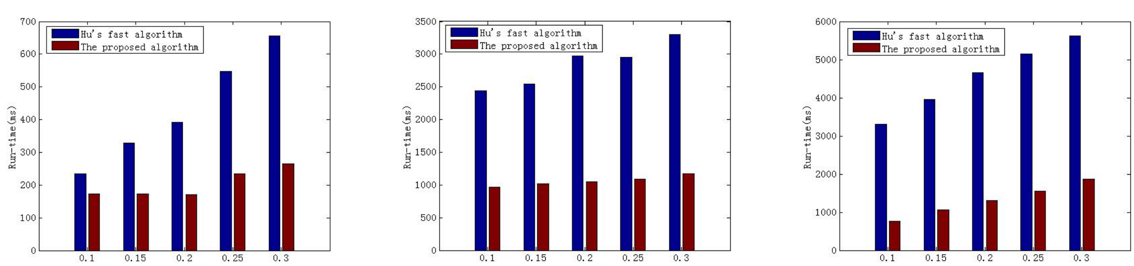
(a) (b) (c)

Fig. 1 Comparisons of run-time on datasets: (a) Credit, (b) Cylinder, (c) German.



(a) (b) (c)

Fig. 2 Comparisons of run-time on datasets: (a) Heart, (b) Hypothyroid, (c) Diabetes.

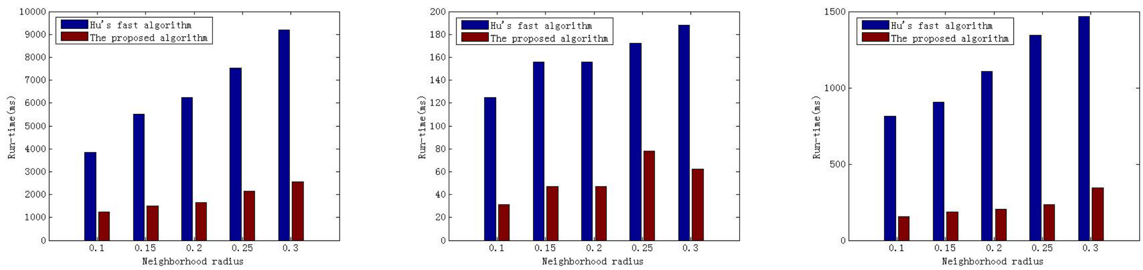


(a) (b) (c)

Fig. 3 Comparisons of run-time on datasets: (a) Image, (b) Ionosphere, (c) Sonar.

6. Conclusions

Traditional neighborhood rough set models employed neighborhoods to construct the theoretical and algorithmic framework. In this paper, we extracted inconsistent objects from traditional neighborhoods to get a new concept called inconsistent neighborhood. Firstly, a number of interesting properties were obtained, which provide some new formulations and some efficient solutions for the theory of neighborhood rough set. Then, the obtained properties were used to design a forward attribute reduction algorithm, which has



(a) (b) (c)

Fig. 4 Comparisons of run-time on datasets: (a) Wdbc, (b) Wine, (c) Wpbc.

been validated to be much more efficient than the existing attribute reduction algorithms in the domain of neighborhood rough set. This demonstrates the advantage of using inconsistent neighborhoods in the applications. The introduction of inconsistent neighborhood concept would provide a new insight into the theory of neighborhood rough set.

To facilitate the comparison between inconsistent neighborhood and traditional neighborhood, the framework of the proposed attribute reduction algorithm is similar with that of Hu's fast reduction algorithm mentioned above. However, if all objects in the universe are unclassifiable by using a single conditional attribute, the reduct obtained by this kind of algorithm will be an empty set. We will study this problem in our future work.

Acknowledgements

This work is supported in part by the Natural Science Foundation of Fujian Province, China under Grant No. 2017J01771, the Education Department of Fujian Province under Grant No. JAT160291, the Institute of Meteorological Big Data-Digital Fujian and Fujian Key Laboratory of Data Science and Statistics.

References

[1] T.-P. Hong, L.-H. Tseng, and B.-C. Chien, "Mining from incomplete quantitative data by fuzzy rough sets," Expert Syst. Appl., vol.37, no.3, pp.2644–2653, 2010.

[2] E.V. Reddy, G.V. Suresh, E.S. Reddy, and S.S. Shaik, "Rough set anlaysis for uncertain data classification," Proc. International Conf. on Trends in Information Sciences and Computing, pp.22–29, 2011.

[3] S.H. Nguyen, A. Skowron, and J. Stepaniuk, "Granular Computing: a Rough Set Approach," Computational Intelligence, vol.17, no.3, pp.514–544, 2001.

[4] Z. Pawlak and A. Skowron, "Rudiments of rough sets," Inf. Sci., vol.177, no.1, pp.3–27, 2007.

[5] Y. Yao and Y. Zhao, "Attribute reduction in decision-theoretic rough set models," Inf. Sci., vol.178, no.17, pp.3356–3373, 2008.

[6] S. Liao, Q. Zhu, and F. Min, "Cost-sensitive attribute reduction in decision-theoretic rough set models," Math. Probl. Eng., vol.2014, Article ID 875918, pp.1–9, 2014.

[7] S. Greco, M. Inuiguchi, and R. Slowinski, "Dominance-based rough set approach using possibility and necessity measures," Proc. Rough Sets and Current Trends in Computing, LNCS, vol.2475, pp.85–92, 2002.

[8] Y. Qian, J. Liang, Y. Yao, and C. Dang, "Mgrs: a multi-granulation rough set," Inf. Sci., vol.180, no.6, pp.949–970, 2010.

[9] G. Lin, J. Liang, and Y. Qian, "Multigranulation rough sets: from partition to covering," Inf. Sci., vol.241, no.12, pp.101–118, 2013.

[10] W.-Z. Wu, J.-S. Mi, and W.-X. Zhang, "Generalized fuzzy rough sets," Inf. Sci., vol.151, pp.263–282, 2003.

[11] H. Chen, T. Li, Y. Cai, C. Luo, and H. Fujita, "Parallel attribute reduction in dominance-based neighborhood rough set," Inf. Sci., vol.373, pp.351–368, 2016.

[12] Y. Qian, J. Liang, W. Pedrycz, and C. Dang, "Positive approximation: An accelerator for attribute reduction in rough set theory," Artif. Intell., vol.174, no.9-10, pp.597–618, 2010.

[13] A. Tan, J. Li, Y. Lin, and G. Lin, "Matrix-based set approximations and reductions in covering decision information systems," Int. J. Approximate Reasoning, vol.59, pp.68–80, 2015.

[14] D. Chen and Y. Yang, "Attribute reduction for heterogeneous data based on the combination of classical and fuzzy rough set models," IEEE Trans. Fuzzy Syst., vol.22, no.5, pp.1325–1334, 2014.

[15] H. Liu, F. Hussain, C.L. Tan, and M. Dash, "Discretization: An enabling technique," Data Mining and Knowledge Discovery, vol.6, no.4, pp.393–423, 2002.

[16] C.-T. Su and J.-H. Hsu, "An extended chi2 algorithm for discretization of real value attributes," IEEE Trans. Knowl. Data Eng., vol.17, no.3, pp.437–441, 2005.

[17] Q. Hu, D. Yu, J.F. Liu, and C. Wu, "Neighborhood rough set based heterogeneous feature subset selection," Inf. Sci., vol.178, no.18, pp.3577–3594, 2008.

[18] Q. Hu, W. Pedrycz, D. Yu, and J. Lang, "Selecting discrete and continuous features based on neighborhood decision error minimization," IEEE Trans. Syst. Man Cybern. Part B Cybern., vol.40, no.1, pp.137–150, 2010.

[19] T.Y. Lin, "Neighborhood systems: A qualitative theory for fuzzy and rough sets," Ed. P. Wang, Advances in Machine Intelligence and Soft Computing, Volume IV, pp.132–155, 1997.

[20] https://archive.ics.uci.edu/ml/datasets.html.

[21] Y. Yao, "A partition model of granular computing," Lecture Notes in Computer Science, vol.3100, pp.232–253, 2004.

[22] D.R. Wilson and T.R. Martinez, "Improved heterogeneous distance functions," Artificial Intelligence Research, vol.6, no.1, pp.1–34, 1996.

[23] Z. Pawlak, Rough sets: Theoretical aspects of reasoning about data, Kluwer Academic Publishers, Boston, 1991.

[24] F. Min and W. Zhu, "Attribute reduction of data with error ranges and test costs," Inf. Sci., vol.211, pp.48–67, 2012.

Shujiao Liao received the M.S. degree in applied mathematics from Shantou University, Shantou, Guangdong, China, in 2006. She is currently working toward the Ph.D. degree with the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, China. And she is meanwhile an associate professor at Minnan Normal University, Zhangzhou, Fujian, China. Her major research interests include granular computing, data mining, and machine learning

Qingxin Zhu received the M.S. degree in applied mathematics from Beijing Institute of Technology University, Beijing, China, in 1984 and the Ph.D. degree in operational research and cybernetics from Ottawa University, Canada, in 1993. From 1993 to 1996, he was a postgraduate research assistant with Ottawa University and Carleton University. In 1997, he was a senior research fellow at Nortel Networks Corporation. Since 1998, he has been a teacher at the University of Electronic Science and Technology of China, and has been a professor since 1999. He is the author of three books, and more than 100 articles. His major research interests include algorithm design, information retrieval, optimizing control and search, etc.

Rui Liang received the M.S. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, Sichuan, China, in 2010. He is currently a Ph.D. candidate at the University of Electronic Science and Technology of China. His research interests include computer vision, machine learning, and video semantic analysis.