International Journal of Computational Intelligence Systems
Vol. **12(2)**, 2019, *pp.* 914–928
DOI: https://doi.org/10.2991/ijcis.d.190807.001; ISSN: 1875-6891; eISSN: 1875-6883
https://www.atlantis-press.com/journals/ijcis/

Dynamic Knowledge Update Using Three-Way Decisions in Dominance-Based Rough Sets Approach While the Object Set Varies

Lei Wang[1,2,*], Min Li[1,2], Jun Ye[1,2], Xiang Yu[1,2], Ziqi Wang[3], Shaobo Deng[1,2]

[1]*College of Information Engineering, Nanchang Institute of Technology, Nanchang, Jiangxi, 330099, China*
[2]*Jiangxi Provincial Key Laboratory of Water Information Cooperative Sensing and Intelligent Processing, Nanchang, Jiangxi, 330099, China*
[3]*Shenzhen HeXunHuaGu Information Technology Co., Ltd. Guangzhou Branch, Guangzhou, Guangdong, 510620, China*

ARTICLE INFO

Article History
Received 06 Jan 2019
Accepted 01 Aug 2019

Keywords
Dominance-based rough sets approach (DRSA)
Three-way decisions (3wds)
Approximations
Dynamic maintenance
Vector inner product

ABSTRACT

Dominance-based rough set approach is the extension of classical Pawlak rough set theories and methodologies, in which the information with preference-ordered relation on the domain of attribute value is fully considered. In the dominance-based information system, upper and lower approximations will be changed while the object set varies over time and the approximations need to be updated correspondingly for their variations result in changes of knowledge and rules. Considering that three-way decisions is a special class of general and effective human ways of problem solving and information processing, a new incremental maintenance mechanism using three-way decisions is proposed in this paper, namely, the universe is divided into three pair-wise disjoint subsets firstly, then appropriate strategies are developed and acted on each subsets. Furthermore, the corresponding methods for updating the approximations of upward unions and downward unions of decision classes are analyzed systematically under the variations of object set in the dominance-based information system from the perspective of three-way decisions. Moreover, considering vector representation and calculation is intuitive and concise, two incremental update algorithms of approximations are suggested and implemented in the MATLAB platform. Finally, some tests on data sets from UCI (UC Irvine Machine Learning Repository) are undertaken to verify the effectiveness of the proposed methods. Compared with the non-incremental updating methods, the proposed incremental updating method with three-way decisions generally exhibits a better performance.

1. INTRODUCTION

Rough set theory (RST) provides a powerful mathematical tool for modeling and processing problems with incomplete, imprecise, uncertain and vague information [1] and it is one of the three primary models of Granular Computing [2] (GrC). The recent two decades have witnessed the booming interest and growing development in research of RST and its applications. As a kind of information processing tools, RST has been widely applied to the fields of artificial intelligence and cognitive sciences, such as pattern recognition [3], knowledge discovery [4,5], decision making [6,7], inductive reasoning [8] and machine learning [9]. In multiple-criteria decision making problems, attributes with preference-order domain (Criteria) constitute an important kind of attribute and should be brought to our great attention. However, only attribute values themselves, by which to tell one object from another, are considered and the preference-order relations among criteria values domain are not considered in RST. On this account, the information with preference-order attribute values domain cannot be processed by RST. Greco *et al.* [10–12] proposed the Dominance-based Rough Set Approach (DRSA) to address this issue, in which the innovation mainly lies in the substitution of the indiscernibility relation by dominance relation. Comparing with the decision rules derived from RST, those derived from DRSA are easy to understand as well as consistent with the practical situation due to the fact that the ordering properties of attributes are taken into account in DRSA. In real-life application, information system (IS) always varies with time for the generation and the collection of data are dynamic, called dynamic IS, it can cause the corresponding changes of the approximations of a concept from which the knowledge are acquired [13]. So updates of the approximations are of great importance for knowledge maintenance and other related tasks. Incremental update scheme is an effective and efficient technique for knowledge update, which enables to acquire new knowledge on the basis of prior knowledge without recalculation from scratch under variation of IS. A great deal of researches have been done in the area of incremental knowledge maintenance under rough sets methodologies. In general, the researches on incremental knowledge update can be fallen into four categories owing to the fact that ISs evolve over time with four levels of variational situations, i.e., (1) incremental knowledge update while the object set varies [14–17]; (2) incremental knowledge update while the attribute set varies [18–23]; (3) incremental knowledge update while the attribute values vary [24–27]; (4) incremental knowledge update under the

*Corresponding author. Email: ezhoulei@163.com

simultaneous variations of object set and attribute set (or object set and attribute value, attribute set and attribute value) [28–32]. All these researches will help decision makers update knowledge from different perspectives in different kinds of ISs. Some achievements based on rough set methodologies under the changed IS are as follows:

Li *et al.* presented a new approach for incrementally updating approximations of a concept under the characteristic relation-based rough sets when the attribute set varies over time for dynamic attribute generalization, which can deal with the case of adding and removing some attributes simultaneously in the IS [14]. Huang *et al.* proposed mechanisms for updating approximations in probabilistic set-valued IS with the variation of attributes [31]. Zhang *et al.* developed the change of approximations in neighborhood decision systems when the object set evolves over time, and proposed two fast incremental algorithms for updating approximations when multiple objects enter into or get out of the neighborhood decision table [15]. Considering dynamic updating of attribute reduction for data vary with time, Jing *et al.* examined an incremental attribute reduction algorithm with a multi-granulation view to maintenance reduct of large-scale data sets dynamically [33,34]. Liu *et al.* proposed some relevant strategies and algorithms for incremental learning knowledge in probabilistic rough sets when attributes evolve with time [22]. Zhang *et al.* suggested an incremental methods for updating rough approximations in interval-valued ISs while attributes set evolves [18]. Luo *et al.* developed an incremental learning technique for hierarchical multi-criteria classification while attribute values vary across different levels of granulations [25]. He further analyzed the updating mechanisms of approximations under the variation of the object set and criteria values respectively in set-valued-ordered IS and set-valued decision system [35]. Zhang *et al.* examined the updating approach of approximations of the concept in set-valued IS by using matrix under the variation of attribute set [36]. Wang *et al.* proposed an incremental matrix method for updating approximations under the variable precision rough set model at insertion or deletion of a single object [37]. Huang *et al.* introduced an incremental mechanisms for updating rough fuzzy approximations with simultaneous variation of objects and attributes set by using matrix operator [31].

The theory of three-way decisions was introduced by Yao in order to model a particular class of human heuristic ways of problem solving and information processing [38,39] and its essential ideas are widely applied in many fields and disciplines. As for the application of three-way decisions in dynamic knowledge update field. Yu *et al.* proposed a new tree-based incremental overlapping clustering method using the three-way decision theory in order to efficiently update the clustering when the data changes [40]. Yang *et al.* proposed a unified model of sequential three-way decisions and multi-level incremental processing for complex problem solving [28,41], then they suggested an unified dynamic framework of decision-theoretic rough sets for incrementally updating three-way probabilistic regions [42].

In this paper, inspired by the trisecting-and-acting model of three-way decisions proposed by Yao, the essential ideas of the trisecting-and-acting model, namely, tri-partition and action are applied to analyze the incremental update mechanisms of approximations with the variation of objects set in dynamic dominance-based IS. Subsequently, the updates of the approximations is realized by using column vector as representation tool as well as computational tool, considering that the column vector is concise and intuitional in representation and operation. Our research motivation is to investigate incremental update of approximations from a new view and to exploit its feasibility and efficiency.

The remainder of the paper is organized as follows: In Section 2, some basic concepts of DRST as well as relevant operations of Boolean column vector are introduced. Under the variation of the object set, the re-judgement of existing relationship between subsets is examined in Section 3, the analyses on incremental update of approximations with three-way decisions and its corresponding illustrated example are presented in Sections 4 and 5, respectively. In Section 6, we propose two incremental algorithms using vector operations for computing approximations based on the preceding analyses. Performance evaluations are illustrated in Section 7. The paper ends with conclusions and further research topics in Section 8.

There are two contributions from our research. One is to provide a three-way decisions perspective on incremental update of approximations in DRSA. The other is an application of Boolean column vector inner product in judging the relationship (inclusion or intersection) between two sets, by which some objects need to be merged to the prior approximations and some other objects need to be removed from the prior approximations during the incremental update process of approximations.

2. PRELIMINARIES

2.1. Symbol Notation

For the sake of convenience, the symbols used in this paper and their corresponding meanings are listed in Table 1.

Table 1 | Description of symbols.

Symbol	Meaning
$\boldsymbol{X}$	The column vector of subset X
$\boldsymbol{D}_P, \boldsymbol{D}_P^T$	Dominance relation matrix and its transposed matrix
$\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right), \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n'^{\geqslant}\right)$	The column vector of $\underline{P}\left(Cl_n^{\geqslant}\right)$ before and after insertion or removal of object
$\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right), \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n'^{\geqslant}\right)$	The column vector of $\overline{P}\left(Cl_n^{\geqslant}\right)$ before and after insertion or removal of object
$\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right), \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n'^{\leqslant}\right)$	The column vector of $\underline{P}\left(Cl_n^{\leqslant}\right)$ before and after insertion or removal of object
$\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right), \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n'^{\leqslant}\right)$	The column vector of $\overline{P}\left(Cl_n^{\leqslant}\right)$ before and after insertion or removal of object
ones(n, 1), ones(1, n)	The n-dimensional row and column vector with all the components are equal to 1
$\lvert\cdot\rvert$	The cardinality of set
$[\boldsymbol{A};\boldsymbol{B}]$	A column vector which is constructed though concatenation of column vector $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{A}[m{:}n]$	Column sub-vector which consist of rows m-n of column vector $\boldsymbol{A}$

2.2. The Basic Concepts of Dominance-Based Rough Set Approach

Some basic concepts of RST and DRSA are briefly reviewed in this subsection [1,10–12].

An IS is a quadruple $IS = \left(U, C \cup \{d\}, V, f\right)$, where $U = \{u_1, u_2, ..., u_n\}$ is a non-empty finite set of objects, called the universe. C is a non-empty finite set of condition attributes and d is a decision attribute with $C \cap \{d\} = \phi$. $V = V_C \cup V_d$, where V is the domain of all attributes, V_C is the domain of all condition attributes and V_d is the domain of the decision attribute d; f is a mapping from $U \times (C \cup \{d\})$ to V.

In an IS, $OB \in 2^U \wedge OB \neq \phi$ and $P \in 2^C \wedge P \neq \phi$. Where 2^U and 2^C is the power set of U and C respectively. If there is a preference (decreasing or increasing) relation on two elements x and y of OB for all $a \in P$, denoted by xD_Py. D_P is a dominance relation on universe U with respect to the attribute set P.

$$D_P = \left\{\left(x, y\right) \in U \times U | f(x, a) \geqslant f\left(y, a\right), \forall a \in P\right\}. \tag{1}$$

In Ref. [10], the granules of knowledge of element x used in DRSA for approximation of the unions $Cl_n^{\geqslant}$ and $Cl_n^{\leqslant}$, namely, the P-dominating set and P-dominated set of element x were defined, respectively, as follows:

$$D_P^+(x) = \left\{y \in U | yD_Px\right\}. \tag{2}$$

$$D_P^-(x) = \left\{y \in U | xD_Py\right\}. \tag{3}$$

The equivalence classes partition of universe U by the decision attribute d is denoted as $U/IND(d)$, called set of decision classes. Let $U/IND(d) = \{Cl_n, n \in T\}, T = \{1, ..., m\}$. m denotes the number of decision classes. $\forall r, s \in T$ *such that* $r > s$, it means the objects from Cl_r are preferred to the objects from Cl_s. In DRSA, an upward union and a downward union of decision classes, denoted as $Cl_n^{\geqslant} = \cup_{n' \geqslant n} Cl_{n'}$ and $Cl_n^{\leqslant} = \cup_{n' \leqslant n} Cl_{n'}$ respectively, are no other than the concepts to be approximated and the definitions of approximation were given in Ref. [10].

Definition 1. *Assume that IS is an information system, $P \subseteq C$, $n \in T$. The definitions of lower and upper approximations of $Cl_n^{\geqslant}$ and $Cl_n^{\leqslant}$ are respectively as follows:*

$$\underline{\mathrm{P}}\left(\mathrm{Cl}_n^{\geqslant}\right) = \left\{x \in U | D_P^+(x) \subseteq Cl_n^{\geqslant}\right\}. \tag{4}$$

$$\overline{\mathrm{P}}\left(\mathrm{Cl}_n^{\geqslant}\right) = \left\{x \in U | D_P^-(x) \cap Cl_n^{\geqslant} \neq \phi\right\}. \tag{5}$$

$$\underline{\mathrm{P}}\left(\mathrm{Cl}_n^{\leqslant}\right) = \left\{x \in U | D_P^-(x) \subseteq Cl_n^{\leqslant}\right\}. \tag{6}$$

$$\overline{\mathrm{P}}\left(\mathrm{Cl}_n^{\leqslant}\right) = \left\{x \in U | D_P^+(x) \cap Cl_n^{\leqslant} \neq \phi\right\}. \tag{7}$$

2.3. Brief Introduction to Three-Way Decisions

In this subsection, we briefly introduce the essential ideas of three-way decisions and trisecting-and-acting model used in this research.

Three-way decisions, proposed by Yao, is a general and effective human heuristic way to problem solving and information processing, their aim is to make fast, low cost and/or high benefit decisions in solving problems with uncertainty and imprecision [38,39,43,44]. The essential ideas of three-way decisions are described in term of a ternary classification according to evaluations of a set of criteria [38]. The two basic tasks of three-way decisions, which gives rise to a trisecting-and-acting model of three-way decisions, are trisecting and acting. Trisecting, namely, tri-partition is to divide a whole into three pair-wise disjoint parts or regions and acting is to develop an appropriate strategy on each part or region [39]. In many real applications, trisecting-and-acting model can turn complexity into simplicity [39] due to its firm cognitive basis and appropriate strategies and it would be a simple, general and flexible model.

2.4. The Boolean Column Vector

The basic concepts of Boolean column vector as well as the operations on Boolean Column vector (including the operations we proposed) are introduced in this subsection [45–47]. Definition 2 is quoted from Ref. [45], Definition 3 is quoted from Ref. [46] and Definition 7 is quoted from Ref. [47]. The vector inner product will be used in analyses on incremental updates of approximations of unions of decision classes, the other operations of vector will be used in the incremental update matrix algorithms of the unions' approximations at insertion or deletion of an object in Section 6.

Definition 2. *Assume that $U = \{u_1, u_2, ..., u_n\}$ is a non-empty finite set of objects, called the universe, and $X \subseteq U$. Then the column vector representation of subset X is as follows:*

$$\boldsymbol{X}_{n\times 1} = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{bmatrix} = \left[\, x_1, x_2, ..., x_n\right]^T. \tag{8}$$

where $x_i = \begin{cases} 1 & u_i \in X \\ 0 & u_i \notin X, \end{cases}$

In Formula (8), the location of object u_i in the universe U is i, denoted as $Loc(u_i, U) = i$.

Example 1. *Assume that $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$ and $A = \{u_1, u_3, u_5, u_7, u_8\}$, $B = \{u_2, u_3, u_4, u_5, u_7\}$, then*

$$\boldsymbol{A} = [a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8]^T = [1, 0, 1, 0, 1, 0, 1, 1]^T.$$

$$\boldsymbol{B} = [b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8]^T = [0, 1, 1, 1, 1, 0, 1, 0]^T.$$

Example 2. *Suppose $U = \{u_2, u_1, u_7, u_4, u_8, u_6, u_3, u_5\}$, then $Loc(u_7, U) = 3$, $Loc(u_8, U) = 5$, $Loc(u_5, U) = 8$.*

2.4.1. Operations on Boolean column vector

Assume set X, Y and Z are subsets of university U and their corresponding n-dimension Boolean column vectors are $\boldsymbol{X}$, $\boldsymbol{Y}$, $\boldsymbol{Z}$ respectively and $\boldsymbol{X} = \left[\, x_1, x_2, ..., x_n\right]^T$, $\boldsymbol{Y} = \left[\, y_1, y_2, ..., y_n\right]^T$, $\boldsymbol{Z} = \left[\, z_1, z_2, ..., z_n\right]^T$.

Definition 3. *Let*

$$[\boldsymbol{X}, \boldsymbol{Y}] = \boldsymbol{X}^T \cdot \boldsymbol{Y} = \sum_{i=1}^{m} x_i \cdot y_i. \tag{9}$$

$[\boldsymbol{X}, \boldsymbol{Y}]$ is called inner product of column vector.

The following formulas are easy to gained from Definition 3.

$$[\boldsymbol{X}, \boldsymbol{Y}] = 0 \Leftrightarrow X \cap Y = \phi. \tag{10}$$

$$[\boldsymbol{X}, \boldsymbol{Y}] = |X| \Leftrightarrow X \subseteq Y. \tag{11}$$

$$[\boldsymbol{X}, \boldsymbol{Y}] = |Y| \Leftrightarrow Y \subseteq X. \tag{12}$$

Vector Inner product can be used to judge the intersection or inclusion between two sets.

Example 3. *(Continuation of Example 1)*

$[\boldsymbol{A}, \boldsymbol{B}] = \sum_{i=1}^{8} a_i \cdot b_i = 3$ *and* $A \cap B \neq \phi$.

Taking into account the characteristics of Boolean column vector, we defined their operations corresponding to the intersection, union and difference of two sets respectively as follows:

Definition 4. *Let*

$$\boldsymbol{Z} = \boldsymbol{X} \cdot * \boldsymbol{Y}. \tag{13}$$

$$\text{where } z_i = \begin{cases} 1 & x_i = 1 \wedge y_i = 1 \\ 0 & \text{others} \end{cases} \qquad i = 1, 2, ..., n$$

$\cdot *$ is called operator of column vector multiplication. It can be used to calculate the intersection of two sets.

Example 4. *(Continuation of Example 1)*

Let $\boldsymbol{D} = \boldsymbol{A} \cdot * \boldsymbol{B} = [0, 0, 1, 0, 1, 0, 1, 0]$. *So* $D = \{u_3, u_5, u_7\}$ *and set D is the intersection of set A and set B.*

Definition 5. *Let*

$$\boldsymbol{Z} = \boldsymbol{X} + \boldsymbol{Y}. \tag{14}$$

$$\text{where } z_i = \begin{cases} 0 & x_i = 0 \wedge y_i = 0 \\ 1 & \text{others} \end{cases} \qquad i = 1, 2, ..., n$$

$\boldsymbol{Z}$ is called summation of $\boldsymbol{X}$ and $\boldsymbol{Y}$. It can be used to calculate the union of two sets.

Example 5. *(Continuation of Example 1)*

Let $\boldsymbol{D} = \boldsymbol{A} + \boldsymbol{B} = [1, 1, 1, 1, 1, 0, 1, 1]$. *So* $D = \{u_1, u_2, u_3, u_4, u_5, u_7, u_8\}$ *and set D is the union of set A and set B.*

Definition 6. Let

$$\boldsymbol{Z} = \boldsymbol{X} - \boldsymbol{Y}. \tag{15}$$

$$\text{where } z_i = \begin{cases} 1 & x_i = 1 \wedge y_i = 0 \\ 0 & \text{others} \end{cases} \qquad i = 1, 2, ..., n$$

$\boldsymbol{Z}$ is called difference of $\boldsymbol{X}$ and $\boldsymbol{Y}$. It can be applied to calculate the difference of two sets.

Example 6. *(Continuation of Example 1)*

Let $\boldsymbol{D} = \boldsymbol{A} - \boldsymbol{B} = [1, 0, 0, 0, 0, 0, 0, 1]$. *So* $D = \{u_1, u_8\}$ *and set D is the subtraction of set A and set B.*

Definition 7. *Assume that X and Y are two non-empty sets, the inclusion degree of X in Y is denote as* $C(X, Y)$.

$$C(X, Y) = \frac{|X \cap Y|}{|X|} = \frac{[\boldsymbol{X}, \boldsymbol{Y}]}{[\boldsymbol{X}, \boldsymbol{X}]}. \tag{16}$$

It is obviously that $0 \leqslant C \leqslant 1$.

Inclusion degree can be used to judge the relationship of two sets. The following formulas are easy to gained from Definitions 7 and 3.

$$C(X, Y) = 0 \Leftrightarrow [\boldsymbol{X}, \boldsymbol{Y}] = 0 \Leftrightarrow X \cap Y = \phi. \tag{17}$$

$$C(X, Y) = 1 \Leftrightarrow [\boldsymbol{X}, \boldsymbol{Y}] = |X| \Leftrightarrow X \subseteq Y. \tag{18}$$

$$C(Y, X) = 1 \Leftrightarrow [\boldsymbol{X}, \boldsymbol{Y}] = |Y| \Leftrightarrow Y \subseteq X. \tag{19}$$

Example 7. *(Continuation of Example 1)*

$$C(A, B) = \frac{|A \cap B|}{|A|} = \frac{[\boldsymbol{A}, \boldsymbol{B}]}{[\boldsymbol{A}, \boldsymbol{A}]} = \frac{3}{5} = 0.6 \text{ and } A \cap B \neq \phi.$$

3. THE RE-JUDGEMENT OF EXISTING RELATIONSHIP BETWEEN SUBSETS UNDER VARIATIONS OF OBJECTS

According to the definitions of approximations, the judgement of inclusion relationship or intersection relationship between two subsets of universe is a prerequisite for calculating approximations in RST. However, the pre-existing relationship between two subsets will change after new objects are inserted into the universe or the original objects are removed from the universe, so the relationship between two subsets remains to be re-judged. In this paper, Boolean column vector is employed to denote the subsets of the universe, and the problem that judge one object belongs to approximations of upward (downward) unions or not can be transformed into the operations of Boolean column vectors and the subsequent simple numeric comparison. In short, the vector approach of dynamic maintenance and update of approximations at variation of objects is that of incremental maintenance and update of approximations via operations on column vectors. Therefore, the research of this section is the foundation of follow-up analyses on incremental update of approximations from perspective of trisecting and acting.

Suppose that set A and set B are two non-empty subsets of the universe U. A' and B' denote updated A and updated B, respectively, after x^+ is inserted into U or x^- is removed from U respectively.

The Lemma 1 and Lemma 2 can be applied to re-judge the relationship between set A and set B after object x^+ is inserted into universe U.

Lemma 1. *Assume that* $A \nsubseteq B$ $\left(i.e., 0 < [\boldsymbol{A}, \boldsymbol{B}] < min\left(|A|, |B|\right)\right)$, *then*

$$A \cap B \neq \phi \Rightarrow A' \cap B' \neq \phi \wedge A' \nsubseteq B'. \tag{20}$$

$$A \cap B = \phi \wedge x^+ \in A' \cap B' \Rightarrow A' \cap B' \neq \phi. \tag{21}$$

The equivalent expressions of column vector are as follows:

$$[\boldsymbol{A}, \boldsymbol{B}] \neq 0 \Rightarrow 0 < \left[\boldsymbol{A}', \boldsymbol{B}'\right] < min\left(|A'|, |B'|\right). \tag{22}$$

$$[\boldsymbol{A}, \boldsymbol{B}] = 0 \wedge x^+ \in \boldsymbol{A}' \cap \boldsymbol{B}' \Rightarrow \left[A', B'\right] = 1. \tag{23}$$

Lemma 2. *Assume that $A \subseteq B$ $\left(i.e., [\boldsymbol{A}, \boldsymbol{B}] = |A|\right)$, the inclusion relationship between A and B will be changed after x^+ is inserted into U if and only if $x^+ \in A' \wedge x^+ \notin B'$.*

$$A \subseteq B \text{ and } x^+ \in A' \wedge x^+ \notin B' \Rightarrow A' \nsubseteq B'. \tag{24}$$

Its equivalent expression of column vector's inner product is as follows:

$$[\boldsymbol{A}, \boldsymbol{B}] = |A| \text{ and } x^+ \in A' \wedge x^+ \notin B' \Rightarrow \left[\boldsymbol{A}', \boldsymbol{B}'\right] \neq |A'|. \tag{25}$$

Proof. The above conclusions can be driven easily from connotation of the inclusion relation between two sets.

Lemma 2 can be demonstrated by the Venn diagram in the Fig. 1.

Similarly, the Lemma 3 and Lemma 4 can be applied to re-judge the relationship between set A and set B after object x^- is removed from universe U.

Lemma 3. *Assume that $A \nsubseteq B$, then*

$$x^- \in A \cap B \Rightarrow A' \nsubseteq B. \tag{26}$$

$$A \cap B = \phi \vee A \cap B = \{x^-\} \Rightarrow A' \cap B' = \phi. \tag{27}$$

The equivalent expressions of column vector are as follows:

$$[\boldsymbol{A}, \boldsymbol{B}] \neq 0 \wedge x^- \notin A \cap B \Rightarrow \left[\boldsymbol{A}', \boldsymbol{B}'\right] \neq 0. \tag{28}$$

$$[\boldsymbol{A}, \boldsymbol{B}] = 0 \vee A \cap B = \{x^-\} \Rightarrow \left[\boldsymbol{A}', \boldsymbol{B}'\right] = 0. \tag{29}$$

Lemma 4. *Assume that $A \subseteq B$, then*

$$A \subseteq B \Rightarrow A' \subseteq B'. \tag{30}$$

Its equivalent expression of column vector's inner product is as follows:

$$[\boldsymbol{A}, \boldsymbol{B}] = |A| \Rightarrow \left[\boldsymbol{A}', \boldsymbol{B}'\right] = |A'|. \tag{31}$$

Proof. The above conclusions can be driven easily from connotation of the inclusion relation. As shown in the Venn diagram in Fig. 2, three cases are considered, i.e., $x^- \in A$, $x^- \in A^C \cap B$ and $x^- \in A^C \cap B^c$.

Lemma 4 demonstrates that the removal of single object from the universe will not alter the existing inclusion relationship between two subsets.

4. THE ANALYSES ON INCREMENTAL UPDATE OF APPROXIMATIONS WITH THREE-WAY DECISIONS

In this section, a trisecting-and-acting model is presented to update approximations in DRSA incrementally and then the methods of tri-partition of universe at insertion or removal of one object are discussed.

4.1. A Trisecting-and-Acting Model for Incremental Update of Approximations in DRSA

The incremental update method of approximations is just the method in which updated approximations can be acquired on the basis of the approximations prior to update by some certain operations and it is not necessary to recompute from scratch. As for the incremental update using three-way decisions, the certain operations refer to trisecting and acting.

We take the update of $\underline{P}\left(Cl_n^{\geq}\right)$ under the insertion of one object x^+ as an example to demonstrate the divide-and-conquer method in trisecting-and-acting model for incremental update of approximations.

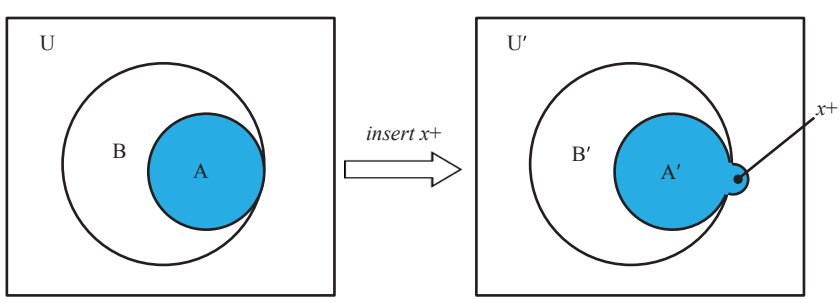

Figure 1 The change of inclusion relation between two subsets after insertion of object x^+.

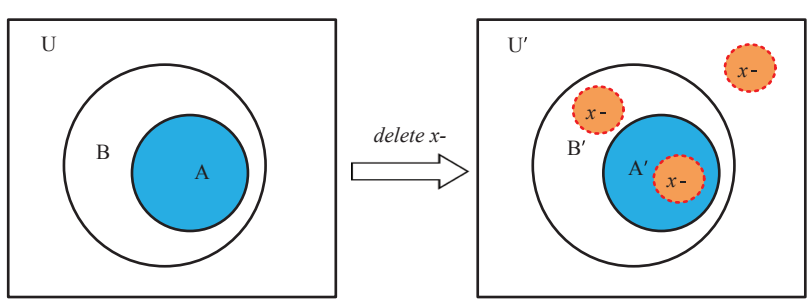

Figure 2 The inclusion relation between two subsets remains unchanged after deletion of object x^-.

Above all, the objects in the universe U' can be partitioned into three pair-wise disjoint parts, as shown in Fig. 3.

The first part consists of all the objects which do not belong to the original approximation and belong to the updated approximation simultaneously, denoted as N, shown in Formula (32).

$$N = \left\{x \in U | x \notin \underline{P}\left(Cl_n^{\geqslant}\right) \wedge x \in \underline{P}\left(Cl_n'^{\geqslant}\right)\right\}. \tag{32}$$

The second part consists of all the objects which belong to the approximation before update and no longer belong to the approximation after update, denoted as M, shown in Formula (33).

$$M = \left\{x \in U | x \in \underline{P}\left(Cl_n^{\geqslant}\right) \wedge x \notin \underline{P}\left(Cl_n'^{\geqslant}\right)\right\}. \tag{33}$$

The rest objects of the universe constitute the third part, i.e., the third part is composed of all the objects except for the first part and the second part, denoted as O and shown in Formula (34).

$$\begin{aligned} O = \{x | & x \notin \underline{P}\left(Cl_n^{\geqslant}\right) \wedge x \notin \underline{P}\left(Cl_n'^{\geqslant}\right) \vee \\ & x \in \underline{P}\left(Cl_n^{\geqslant}\right) \wedge x \in \underline{P}\left(Cl_n'^{\geqslant}\right)\}. \end{aligned} \tag{34}$$

It's worth noting that the inserted object x^+ is a special object. x^+ may be belongs to either the set N or the set O and we need to determine which set x^+ belong to.

Secondly, the different strategy will be developed and adopted in the different part after trisection. It is obviously that the strategies should be developed and adopted only in the set N and the set M after tri-partition of universe.

So the incremental update of $\underline{P}\left(Cl_n^{\geqslant}\right)$ can be finished by the following three steps:

1. Computes the set M and the set N.
2. Subtracts the set M from $\underline{P}\left(Cl_n^{\geqslant}\right)$, assume that the intermediate result is $\underline{P}''\left(Cl_n^{\geqslant}\right)$.
3. Computes the union set of $\underline{P}''\left(Cl_n^{\geqslant}\right)$ and the set N, so the updated approximation $\underline{P}\left(Cl_n'^{\geqslant}\right)$ is gained.

Thus, the incremental update of $\underline{P}\left(Cl_n^{\geqslant}\right)$ is more appropriately formulated as the following formula:

$$\underline{P}\left(Cl_n'^{\geqslant}\right) = \underline{P}\left(Cl_n^{\geqslant}\right) - M \cup N. \tag{35}$$

The incremental update of $\underline{P}\left(Cl_n^{\geqslant}\right)$ follows Formula (35) after tri-partition of universe. And the incremental update of $\overline{P}\left(Cl_n^{\geqslant}\right)$, $\underline{P}\left(Cl_n^{\leqslant}\right)$ and $\overline{P}\left(Cl_n^{\leqslant}\right)$ are similar with that of $\underline{P}\left(Cl_n^{\geqslant}\right)$.

As for the case of object's deletion, the process of incremental update of approximations is similar with that of object's insertion, so it is not discussed again.

In the following subsections, under the unified frame of trisecting-and-acting model, only methods of tri-partition are discussed for update on dynamic environment of insertion and removal of one object, respectively. What's more, it can be simplified as a series of calculational formulas of the set M and N. The inner product of Boolean column vectors is employed to judge the relationship between two sets in the proof of the theorems.

4.2. The Update of Approximations Using Trisecting-and-Acting Model at Insertion of Object

Supposing x^+ is the inserted object and $Loc\left(x^+, U\right) = i$.

4.2.1. The update of unions of decision classes

Supposing that Cl_n' is the decision class to which inserted object x^+ belongs.

1. The update of upward unions of decision classes
   The updated upper unions of decision classes is as following formula:

$$Cl_n'^{\geqslant} = \begin{cases} Cl_n^{\geqslant} \cup \{x^+\} & n' \geqslant n \\ Cl_n^{\geqslant} & n' < n \end{cases}. \tag{36}$$

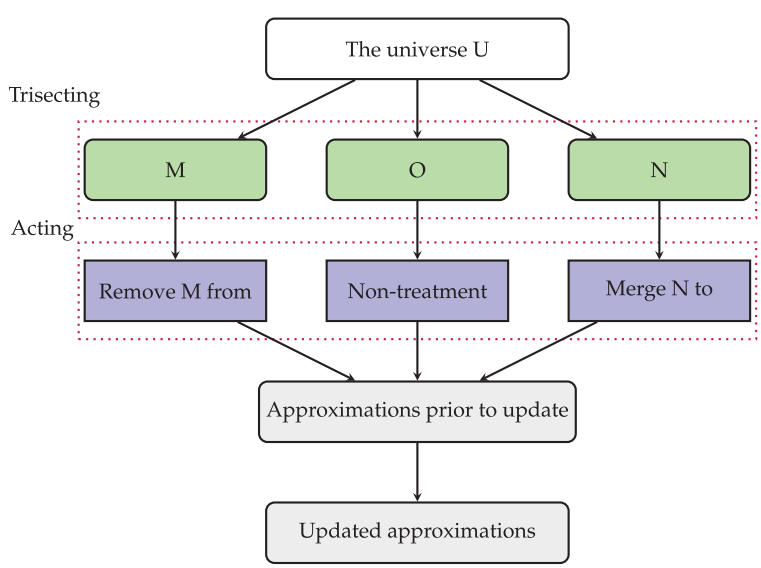

Figure 3 | The unified framework of update of approximations.

The corresponding column vector of the updated upward unions of decision classes can be denoted as

$$\boldsymbol{Cl}_n^{\prime\geqslant} = \begin{cases} \left[\boldsymbol{Cl}_n^{\geqslant}(1:i-1);1;\boldsymbol{Cl}_n^{\geqslant}(i+1:m)\right] & n' \geqslant n \\ \left[\boldsymbol{Cl}_n^{\geqslant}(1:i-1);0;\boldsymbol{Cl}_n^{\geqslant}(i+1:m)\right] & n' < n \end{cases}. \quad (37)$$

2. The update of downward unions of decision classes
The updated downward unions of decision classes is as following formula:

$$Cl_n^{\prime\leqslant} = \begin{cases} Cl_n^{\leqslant} \cup \{x^+\} & n' \leqslant n \\ Cl_n^{\leqslant} & n' > n \end{cases}. \quad (38)$$

The corresponding column vector of the updated upward union set can be denoted as

$$\boldsymbol{Cl}_n^{\prime\leqslant} = \begin{cases} \left[\boldsymbol{Cl}_n^{\leqslant}(1:i-1);1;\boldsymbol{Cl}_n^{\leqslant}(i+1:m)\right] & n' \leqslant n \\ \left[\boldsymbol{Cl}_n^{\leqslant}(1:i-1);0;\boldsymbol{Cl}_n^{\leqslant}(i+1:m)\right] & n' > n \end{cases}. \quad (39)$$

It is obviously that the dimension of updated column vector will increase to n + 1 after removal of x^+.

4.2.2. Trisection of universe at insertion of object

The calculations of the set N and M are discussed in Theorems 1–4 respectively. The computational results lead to tri-partition of universe naturally.

- The computational formula of the set M and N for incremental update of $\underline{P}\left(Cl_n^{\geqslant}\right)$

Theorem 1.

$$M = \begin{cases} \phi & n' \geqslant n \\ \left\{u | u \in D_P'^{-}\left(x^+\right) \wedge u \in \underline{P}\left(Cl_n^{\geqslant}\right)\right\} & n' < n \end{cases}. \quad (40)$$

$$N = \begin{cases} \{x^+\} & n' \geqslant n \wedge D_P'^{+}\left(x^+\right) \subseteq Cl_n'^{\geqslant} \\ \phi & n' < n \end{cases}. \quad (41)$$

Proof.
$n' \geqslant n$: $x^+ \in Cl_n'^{\geqslant}$.
Case 1: If $D_P'^{+}\left(x^+\right) \subseteq Cl_n'^{\geqslant}$, i.e.,
$\left[\boldsymbol{Cl}_n'^{\geqslant}\boldsymbol{D}_{\mathbf{P}}'^{+}\left(x^+\right)\right] = \left|D_P'^{+}\left(x^+\right)\right|$, we have $x^+ \in \underline{P}\left(Cl_n'^{\geqslant}\right)$.
Case 2: Assume that $v \notin \underline{P}\left(Cl_n^{\geqslant}\right)$, then $D_P^{+}(v) \nsubseteq Cl_n^{\geqslant}$, we have $\left[\boldsymbol{Cl}_n^{\geqslant}, \boldsymbol{D}_{\mathbf{P}}^{+}(v)\right] \neq \left|D_P^{+}(v)\right|$, and $\left[\boldsymbol{Cl}_n'^{\geqslant}, \boldsymbol{D}_{\mathbf{P}}'^{+}(v)\right] \neq \left|D_P'^{+}(v)\right|$ still holds after x^+ is inserted into U (Lemma 1).
To sum up, $N = \{x^+\}$ while $D_P'^{+}\left(x^+\right) \subseteq Cl_n'^{\geqslant}$.
Case 3: Assume that $u \in \underline{P}\left(Cl_n^{\geqslant}\right)$, then $D_P^{+}(v) \subseteq Cl_n^{\geqslant}$, we have $\left[\boldsymbol{Cl}_n^{\geqslant}, \boldsymbol{D}_{\mathbf{P}}^{+}(u)\right] = \left|D_P^{+}(u)\right|$, For
$x^+ \in Cl_n'^{\geqslant}$, $\left[\boldsymbol{Cl}_n'^{\geqslant}, \boldsymbol{D}_{\mathbf{P}}'^{+}(u)\right] = \left|D_P'^{+}(u)\right|$ still holds after x^+ is inserted into U (Lemma 2).
So, $M = \phi$.
$n' < n$:
Case 1: For $x^+ \notin Cl_n'^{\geqslant}$ and $x^+ \in D_P'^{+}\left(x^+\right)$, we have $\left[Cl_n'^{\geqslant} D_P'^{+}\left(x^+\right)\right] \neq \left|D_P'^{+}\left(x^+\right)\right|$, so $x^+ \notin N$.
Case 2: Assume that $v \notin \underline{P}\left(Cl_n^{\geqslant}\right)$, then $D_P^{+}(v) \nsubseteq Cl_n^{\geqslant}$, we have $\left[\boldsymbol{Cl}_n^{\geqslant}, \boldsymbol{D}_{\mathbf{P}}^{+}(v)\right] \neq \left|D_P^{+}(v)\right|$, and $\left[\boldsymbol{Cl}_n'^{\geqslant}\boldsymbol{D}_{\mathbf{P}}'^{+}(v)\right] \neq \left|D_P'^{+}(v)\right|$ still holds after x^+ is inserted into U (Lemma 1).
So, $N = \phi$.
Case 3: Assume that $u \in \underline{P}\left(Cl_n^{\geqslant}\right)$, then $D_P^{+}(u) \subseteq Cl_n^{\geqslant}$, we have $\left[\boldsymbol{Cl}_n^{\geqslant}, \boldsymbol{D}_{\mathbf{P}}^{+}(u)\right] = \left|D_P^{+}(u)\right|$. For $x^+ \notin Cl_n'^{\geqslant}$ and if $x^+ \in D_P'^{+}(u)$, i.e., $u \in D_P'^{-}\left(x^+\right)$, then $\left[\boldsymbol{Cl}_n'^{\geqslant}\boldsymbol{D}_{\mathbf{P}}'^{+}(u)\right] \neq \left|D_P'^{+}(u)\right|$ (Lemma 2).
So, $M = \left\{u | u \in D_P'^{-}\left(x^+\right) \wedge u \in \underline{P}\left(Cl_n^{\geqslant}\right)\right\}$.

- The computational formula of the set M and N for incremental update of $\overline{P}\left(Cl_n^{\geqslant}\right)$

Theorem 2.

$$M = \phi. \quad (42)$$

$$N = \begin{cases} \left\{v | v \in D_P'^{+}\left(x^+\right) \wedge v \notin \overline{P}\left(Cl_n^{\geqslant}\right)\right\} & n' \geqslant n \\ \{x^+\} & n' < n \wedge D_P'^{-}\left(x^+\right) \cap Cl_n'^{\geqslant} \neq \phi. \end{cases} \quad (43)$$

Proof.
Assume that $v \in \overline{P}\left(Cl_n^{\geqslant}\right)$, then $D_P^{-}(v) \cap Cl_n^{\geqslant} \neq \phi$, and $D_P'^{-}(v) \cap Cl_n'^{\geqslant} \neq \phi$ still holds after x^+ is inserted (Lemma 1).
So, $M = \phi$.
$n' \geqslant n : x^+ \in Cl_n'^{\geqslant}$.
Case 1: For $x^+ \in D_P^{-}\left(x^+\right)$, we have $D_P'^{-}\left(x^+\right) \cap Cl_n'^{\geqslant} \neq \phi$, i.e., $\left[\boldsymbol{D}_{\mathbf{P}}'^{-}\left(x^+\right), \boldsymbol{Cl}_n'^{\geqslant}\right] \neq 0$, so $x^+ \in N$.
Case 2: Assume that $v \notin \overline{P}\left(Cl_n^{\geqslant}\right)$, then $D_P^{-}(v) \cap Cl_n^{\geqslant} = \phi$. For $x^+ \in Cl_n'^{\geqslant}$, and if $x^+ \in D_P'^{-}(v)$, i.e., $v \in D_P'^{+}\left(x^+\right)$, then $D_P'^{-}(v) \cap Cl_n'^{\geqslant} \neq \phi$, thus $v \in \overline{P}\left(Cl_n'^{\geqslant}\right)$ (Lemma 1).
So, $N = \left\{v | v \notin \overline{P}\left(Cl_n^{\geqslant}\right) \wedge v \in D_P^{+}\left(x^+\right)\right\} \cup \{x^+\} = \left\{v | v \notin \overline{P}\left(Cl_n^{\geqslant}\right) \wedge v \in D_P^{+}\left(x^+\right)\right\}$.
$n' < n : x^+ \notin Cl_n'^{\geqslant}$
Case 1: If $D_P'^{-}\left(x^+\right) \cap Cl_n'^{\geqslant} \neq \phi$, i.e., $\left[\boldsymbol{D}_{\mathbf{P}}'^{-}\left(x^+\right), \boldsymbol{Cl}_n'^{\geqslant}\right] \neq 0$, then $x^+ \in N$.
Case 2: Assume that $v \notin \overline{P}\left(Cl_n^{\geqslant}\right)$, then $D_P^{-}(v) \cap Cl_n^{\geqslant} = \phi$. For $x^+ \notin Cl_n'^{\geqslant}$, no matter whether $x^+ \in D_P'^{-}(v)$ or not, $D_P'^{-}(v) \cap Cl_n'^{\geqslant} = \phi$ still holds after x^+ is inserted into U (Lemma 1), thus, $v \notin \overline{P}\left(Cl_n'^{\geqslant}\right)$.
To sum up, $N = \{x^+\}$ when $D_P'^{-}\left(x^+\right) \cap Cl_n'^{\geqslant} \neq \phi$.

- The computational formula of the set M and N for incremental update of $\underline{P}\left(Cl_n^{\leqslant}\right)$

Theorem 3.

$$M = \begin{cases} \left\{u | u \in D_P'^{+}\left(x^+\right) \wedge u \in \underline{P}\left(Cl_n^{\leqslant}\right)\right\} & n' > n \\ \phi & n' \leqslant n \end{cases}. \quad (44)$$

$$N = \begin{cases} \phi & n' > n \\ \{x^+\} & n' \leqslant n \wedge D_P'^{-}\left(x^+\right) \subseteq Cl_n'^{\leqslant} \end{cases}. \quad (45)$$

Proof.
The proof of Theorem 3 is similar to that of Theorem 1.

- The computational formula of the set M and N for incremental update of $\overline{P}\left(Cl_n^{\leqslant}\right)$

Theorem 4.

$$M = \phi. \tag{46}$$

$$N = \begin{cases} \{x^+\} & n' > n \wedge D_P'^{+}\left(x^+\right) \cap Cl_n'^{\leqslant} \neq \phi \\ \left\{v | v \in D_P'^{-}\left(x^+\right) \wedge v \notin \overline{P}\left(Cl_n^{\leqslant}\right)\right\} & n' \leqslant n. \end{cases} \tag{47}$$

Proof.
The proof of Theorem 4 is similar to that of Theorem 2.

4.3. The Update of Approximations Using Trisecting-and-Acting Model at Deletion of Object

Supposing x^- is the deleted object and $Loc(x^-, U) = i$.

4.3.1. The update of unions of decision classes

Supposing that $Cl_{n'}$ is the decision class to which x^- belongs.

1. The update of upward unions of decision classes
The updated upward unions of decision classes is as following formula:

$$Cl_n'^{\geqslant} = \begin{cases} Cl_n^{\geqslant} - \{x^-\} & n' \geqslant n \\ Cl_n^{\geqslant} & n' < n \end{cases}. \tag{48}$$

The corresponding column vector of the updated upward union set can be denoted as

$$\boldsymbol{Cl}_n'^{\geqslant} = \left[\boldsymbol{Cl}_n^{\geqslant}(1 : i-1); \boldsymbol{Cl}_n^{\geqslant}(i+1 : m)\right]. \tag{49}$$

2. The update of downward unions of decision classes
The updated downward unions of decision classes is as following formula:

$$Cl_n'^{\leqslant} = \begin{cases} Cl_n^{\leqslant} - \{x^-\} & n' \leqslant n \\ Cl_n^{\leqslant} & n' > n \end{cases}. \tag{50}$$

The corresponding column vector of the updated upward unions can be denoted as

$$\boldsymbol{Cl}_n'^{\leqslant} = \left[\boldsymbol{Cl}_n^{\leqslant}(1 : i-1); \boldsymbol{Cl}_n^{\leqslant}(i+1 : m)\right]. \tag{51}$$

It is obviously that the dimension of updated column vector will decrease to n−1 after removal of x^-.

4.3.2. Trisection of universe at deletion of object

The calculations of the set N and M are discussed in Theorem 5–8, respectively. The computational results lead to tri-partition of universe naturally.

- The computational formula of the set M and N for incrementally update of $\underline{P}(Cl_n^{\geqslant})$

Theorem 5.

$$M = \begin{cases} \{x^-\} & n' \geqslant n \wedge x^- \in \underline{P}\left(Cl_n^{\geqslant}\right) \\ \phi & n' < n \end{cases}. \tag{52}$$

$$N = \begin{cases} \phi & n' \geqslant n \\ \left\{v | v \notin \underline{P}\left(Cl_n^{\geqslant}\right) \wedge v \in D_P^-(x^-) \right. & \\ \left. \wedge D_P'^{+}(v) \subseteq Cl_n'^{\geqslant}\right\} & n' < n \end{cases}. \tag{53}$$

Proof.

$$\begin{aligned} M &= \left\{u | u \in \underline{P}\left(Cl_n^{\geqslant}\right) \wedge u \notin \underline{P}\left(Cl_n'^{\geqslant}\right)\right\} \\ &= \left\{u | D_P^+(u) \subseteq Cl_n^{\geqslant} \wedge D_P'^{+}(u) \nsubseteq Cl_n'^{\geqslant}\right\}. \end{aligned}$$

$u \in \underline{P}\left(Cl_n'^{\geqslant}\right)$ will still hold after x^- is removed (Lemma 4).
When $n' \geqslant n$, we have $x^- \in Cl_n^{\geqslant}$. If $x^- \in \underline{P}\left(Cl_n^{\geqslant}\right)$, then $M = \{x^-\}$.

$$\begin{aligned} N &= \left\{v | v \notin \underline{P}\left(Cl_n^{\geqslant}\right) \wedge u \in \underline{P}\left(Cl_n'^{\geqslant}\right)\right\} \\ &= \left\{v | D_P^+(v) \nsubseteq Cl_n^{\geqslant} \wedge D_P'^{+}(v) \subseteq Cl_n'^{\geqslant}\right\}. \end{aligned}$$

When $n' \geqslant n$, we have $x^- \in Cl_n^{\geqslant}$. $D_P'^{+}(v) \nsubseteq Cl_n'^{\geqslant}$ will still hold after x^- is removed (lemma 3), so $N = \phi$.
When $n' < n$, we have $x^- \notin Cl_n^{\geqslant}$. If $x^- \in D_P^+(v)$, *i.e.*, $v \in D_P^-(x^-)$ and $D_P'^{+}(v) \subseteq Cl_n'^{\geqslant}$, then $v \in \underline{P}\left(Cl_n'^{\geqslant}\right)$.
So $N = \left\{v | v \notin \underline{P}\left(Cl_n^{\geqslant}\right) \wedge v \in D_P^-(x^-) \wedge D_P'^{+}(v) \subseteq Cl_n'^{\geqslant}\right\}$.

- The computational formula of the set M and N for incrementally update of $\overline{P}\left(Cl_n^{\geqslant}\right)$

Theorem 6.

$$M = \begin{cases} \{x^-\} & n' < n \\ & \wedge x^- \in \overline{P}\left(Cl_n^{\geqslant}\right) \\ \{x^-\} \cup \left\{u | u \in \overline{P}\left(Cl_n^{\geqslant}\right) \cap \right. & \\ \left. D_P^+(x^-) \wedge D_P'^{-}(u) \cap Cl_n'^{\geqslant} = \phi\right\} & n' \geqslant n \end{cases}. \tag{54}$$

$$N = \phi. \tag{55}$$

Proof.

$$\begin{aligned} M &= \left\{u | u \in \overline{P}\left(Cl_n^{\geqslant}\right) \wedge u \notin \overline{P}\left(Cl_n'^{\geqslant}\right)\right\} \\ &= \left\{u | D_P^-(u) \cap Cl_n^{\geqslant} \neq \phi \wedge D_P'^{-}(u) \cap \right. \\ &\quad \left. Cl_n'^{\geqslant} = \phi\right\}. \end{aligned}$$

When $n' \geqslant n$, we have $x^- \in Cl_n^{\geqslant}$, so $D_P^-(x^-) \cap Cl_n^{\geqslant} \neq \phi$, i.e., $x^- \in M$. And if $u \in D_P^+(x^-)$, i.e., $x^- \in D_P^-(u)$ and $D_P'^{-}(u) \cap Cl_n'^{\geqslant} = \phi$, then $u \notin \overline{P}\left(Cl_n'^{\geqslant}\right)$.
To sum up, $M = \{x^-\} \cup \left\{u | u \in \overline{P}\left(Cl_n^{\geqslant}\right) \cap D_P^+(x^-) \wedge D_P'^{-}(u) \cap Cl_n'^{\geqslant} = \phi\right\}$

When $n' < n$, we have $x^- \notin Cl_n^{\geqslant}$, $D_P'^-(u) \cap Cl_n'^{\geqslant} \neq \phi$ still hold after x^- is removed. If $D_P^-(x^-) \cap Cl_n^{\geqslant} \neq \phi$, then $M = \{x^-\}$.

$$N = \left\{v | v \notin \overline{P}\left(Cl_n^{\geqslant}\right) \wedge u \in \overline{P}\left(Cl_n'^{\geqslant}\right)\right\}$$
$$= \left\{v | D_P^-(v) \cap Cl_n^{\geqslant} = \phi \wedge D_P'^-(v) \cap Cl_n'^{\geqslant} \neq \phi\right\}.$$

For $D_P'^-(v) \cap Cl_n'^{\geqslant} = \phi$ will still hold after x^- is removed (Lemma 3). So, $N = \phi$.

- The computational formula of the set M and N for incrementally update of $\underline{P}\left(Cl_n^{\leqslant}\right)$

Theorem 7.

$$M = \begin{cases} \{x^-\} & n' \leqslant n \wedge x^- \in \underline{P}\left(Cl_n^{\leqslant}\right) \\ \phi & n' > n \end{cases}. \tag{56}$$

$$N = \begin{cases} \phi & n' \leqslant n \\ \left\{v | v \notin \underline{P}\left(Cl_n^{\leqslant}\right) \wedge v \in D_P^+(x^-) \right. & \\ \left. \wedge D_P'^-(v) \subseteq Cl_n'^{\leqslant}\right\} & n' > n \end{cases}. \tag{57}$$

Proof.
The proof of Theorem 7 is similar to that of Theorem 5.

- The computational formula of the set M and N for incremental update of $\overline{P}\left(Cl_n^{\leqslant}\right)$

Theorem 8.

$$M = \begin{cases} \{x^-\} \cup \left\{u | u \in \overline{P}\left(Cl_n^{\leqslant}\right) \cap \right. & \\ \left. D_P^-(x^-) \wedge D_P'^+(u) \cap Cl_n'^{\leqslant} = \phi\right\} & n' \leqslant n \\ \{x^-\} & n' > n \\ & \wedge x^- \in \overline{P}\left(Cl_n^{\leqslant}\right). \end{cases} \tag{58}$$

$$N = \phi. \tag{59}$$

Proof.
The proof of Theorem 8 is similar to that of Theorem 6.

5. ILLUSTRATIVE EXAMPLE

Consider the following example (See Table 2). A set of 16 objects is described by the set of 3 attributes (criteria) $C = \{c_1, c_2, c_3\}$. The decision attribute d classifies objects into three decision classes Cl_1, Cl_2 and Cl_3 which are preference-ordered according to increasing class number.

$$\begin{aligned} Cl_1^{\leqslant} &= \{o_1, o_2, o_3, o_{10}, o_{14}\}. \\ Cl_2^{\leqslant} &= \{o_1, o_2, o_3, o_4, o_5, o_6, o_9, o_{10}, o_{12}, o_{13}, o_{14}, o_{15}\}. \\ Cl_2^{\geqslant} &= \{o_4, o_5, o_6, o_7, o_8, o_9, o_{11}, o_{12}, o_{13}, o_{15}, o_{16}\}. \\ Cl_3^{\geqslant} &= \{o_7, o_8, o_{11}, o_{16}\}. \end{aligned}$$

Let $P = C$, the approximations of upward and downward unions of decision classes are

$$\begin{aligned} \underline{P}\left(Cl_1^{\leqslant}\right) &= \{o_1, o_2, o_{14}\}. \\ \overline{P}\left(Cl_1^{\leqslant}\right) &= \{o_1, o_2, o_3, o_{10}, o_{14}, o_{15}\}. \\ \underline{P}\left(Cl_2^{\leqslant}\right) &= \{o_1, o_2, o_3, o_5, o_6, o_9, o_{10}, o_{12}, o_{13}, o_{14}, o_{15}\}. \\ \overline{P}\left(Cl_2^{\leqslant}\right) &= \{o_1, o_2, o_3, o_4, o_5, o_6, o_9, o_{10}, o_{11}, o_{12}, o_{13}, \\ &\quad o_{14}, o_{15}\}. \\ \underline{P}\left(Cl_2^{\geqslant}\right) &= \{o_4, o_5, o_6, o_7, o_8, o_9, o_{11}, o_{12}, o_{13}, o_{16}\}. \\ \overline{P}\left(Cl_2^{\geqslant}\right) &= \{o_3, o_4, o_5, o_6, o_7, o_8, o_9, o_{10}, o_{11}, o_{12}, o_{13}, \\ &\quad o_{15}, o_{16}\}. \\ \underline{P}\left(Cl_3^{\geqslant}\right) &= \{o_7, o_8, o_{16}\}. \\ \overline{P}\left(Cl_3^{\geqslant}\right) &= \{o_4, o_7, o_8, o_{11}, o_{16}\}. \end{aligned}$$

Now we consider the following two cases at time $t + 1$:

- A new object o_{17} inserts the IS (See Table 3).

$$\begin{aligned} Cl_1^{\leqslant\prime} &= \{o_1, o_2, o_3, o_{10}, o_{14}\}. \\ Cl_2^{\leqslant\prime} &= \{o_1, o_2, o_3, o_4, o_5, o_6, o_9, o_{10}, o_{12}, o_{13}, o_{14}, o_{15}, o_{17}\}. \\ Cl_2^{\geqslant\prime} &= \{o_4, o_5, o_6, o_7, o_8, o_9, o_{11}, o_{12}, o_{13}, o_{15}, o_{16}, o_{17}\}. \\ Cl_3^{\geqslant\prime} &= \{o_7, o_8, o_{11}, o_{16}\}. \end{aligned}$$

Table 2 Illustrative data table.

Object	c_1	c_2	c_3	d
o_1	0.8	0.5	1.5	Cl_1
o_2	1	1.5	8	Cl_1
o_3	1	3	5	Cl_1
o_4	2.5	4	11	Cl_2
o_5	1.2	1	7	Cl_2
o_6	1.9	4.5	14	Cl_2
o_7	2.8	5.5	15	Cl_3
o_8	2.4	4	13	Cl_3
o_9	0.6	3	6	Cl_2
o_{10}	2	2.5	6	Cl_1
o_{11}	2.4	3.5	9	Cl_3
o_{12}	1.6	3	12	Cl_2
o_{13}	1.8	5	9.5	Cl_2
o_{14}	0.6	2	2.5	Cl_1
o_{15}	1	2	4.5	Cl_2
o_{16}	3	4.5	9	Cl_3

Table 3 Insertion of one object.

Object	c_1	c_2	c_3	d
o_1	0.8	0.5	1.5	Cl_1
o_2	1	1.5	8	Cl_1
o_3	1	3	5	Cl_1
o_4	2.5	4	11	Cl_2
o_5	1.2	1	7	Cl_2
o_6	1.9	4.5	14	Cl_2
o_7	2.8	5.5	15	Cl_3
o_8	2.4	4	13	Cl_3
o_9	0.6	3	6	Cl_2
o_{10}	2	2.5	6	Cl_1
o_{11}	2.4	3.5	9	Cl_3
o_{12}	1.6	3	12	Cl_2
o_{13}	1.8	5	9.5	Cl_2
o_{14}	0.6	2	2.5	Cl_1
o_{15}	1	2	4.5	Cl_2
o_{16}	3	4.5	9	Cl_3
o_{17}	1.8	2.8	3.5	Cl_2

The $D_P'^{-}(o_{17})$ and $D_P'^{+}(o_{17})$ should be calculated firstly.

$$D_P'^{-}(o_{17}) = \{o_1, o_{14}, o_{17}\}.$$
$$D_P'^{+}(o_{17}) = \{o_4, o_6, o_7, o_8, o_{11}, o_{13}, o_{16}, o_{17}\}.$$

According to Theorems 1–4, we have

$$\begin{aligned}\underline{P}\left(Cl_1^{\leqslant\prime}\right) &= \underline{P}\left(Cl_1^{\leqslant}\right) - M \cup N \\ &= \underline{P}\left(Cl_1^{\leqslant}\right) - \phi \cup \phi \\ &= \{o_1, o_2, o_{14}\}.\end{aligned}$$

$$\begin{aligned}\overline{P}\left(Cl_1^{\leqslant\prime}\right) &= \overline{P}\left(Cl_1^{\leqslant}\right) - M \cup N \\ &= \overline{P}\left(Cl_1^{\leqslant}\right) - \phi \cup \phi \\ &= \{o_1, o_2, o_3, o_{10}, o_{14}, o_{15}\}.\end{aligned}$$

$$\begin{aligned}\underline{P}\left(Cl_2^{\leqslant\prime}\right) &= \underline{P}\left(Cl_2^{\leqslant}\right) - M \cup N \\ &= \underline{P}\left(Cl_2^{\leqslant}\right) - \phi \cup \{o_{17}\} \\ &= \{o_1, o_2, o_3, o_5, o_6, o_9, o_{10}, o_{12}, o_{13}, o_{14}, o_{15}, o_{17}\}.\end{aligned}$$

$$\begin{aligned}\overline{P}\left(Cl_2^{\leqslant\prime}\right) &= \overline{P}\left(Cl_2^{\leqslant}\right) - M \cup N \\ &= \overline{P}\left(Cl_2^{\leqslant}\right) - \phi \cup \{o_{17}\} \\ &= \{o_1, o_2, o_3, o_4, o_5, o_6, o_9, o_{10}, o_{11}, o_{12}, o_{13}, o_{14}, o_{15}, o_{17}\}.\end{aligned}$$

$$\begin{aligned}\underline{P}\left(Cl_2^{\geqslant\prime}\right) &= \underline{P}\left(Cl_2^{\geqslant}\right) - M \cup N \\ &= \underline{P}\left(Cl_2^{\geqslant}\right) - \phi \cup \{o_{17}\} \\ &= \{o_4, o_5, o_6, o_7, o_8, o_9, o_{11}, o_{12}, o_{13}, o_{16}, o_{17}\}.\end{aligned}$$

$$\begin{aligned}\overline{P}\left(Cl_2^{\geqslant\prime}\right) &= \overline{P}\left(Cl_2^{\geqslant}\right) - M \cup N \\ &= \overline{P}\left(Cl_2^{\geqslant}\right) - \phi \cup \{o_{17}\} \\ &= \{o_3, o_4, o_5, o_6, o_7, o_8, o_9, o_{10}, o_{11}, o_{12}, o_{13}, o_{15}, \\ &\quad o_{16}, o_{17}\}.\end{aligned}$$

$$\begin{aligned}\underline{P}\left(Cl_3^{\geqslant\prime}\right) &= \underline{P}\left(Cl_3^{\geqslant}\right) - M \cup N \\ &= \underline{P}\left(Cl_3^{\geqslant}\right) - \phi \cup \phi \\ &= \{o_7, o_8, o_{16}\}.\end{aligned}$$

$$\begin{aligned}\overline{P}\left(Cl_3^{\geqslant\prime}\right) &= \overline{P}\left(Cl_3^{\geqslant}\right) - M \cup N \\ &= \overline{P}\left(Cl_3^{\geqslant}\right) - \phi \cup \phi \\ &= \{o_4, o_7, o_8, o_{11}, o_{16}\}.\end{aligned}$$

- The object o_{10} gets out of the IS (See Table 4).

$$\begin{aligned}Cl_1^{\leqslant\prime} &= \{o_1, o_2, o_3, o_{14}\}. \\ Cl_2^{\leqslant\prime} &= \{o_1, o_2, o_3, o_4, o_5, o_6, o_9, o_{12}, o_{13}, o_{14}, o_{15}\}. \\ Cl_2^{\geqslant\prime} &= \{o_4, o_5, o_6, o_7, o_8, o_9, o_{11}, o_{12}, o_{13}, o_{15}, o_{16}\}. \\ Cl_3^{\geqslant\prime} &= \{o_7, o_8, o_{11}, o_{16}\}.\end{aligned}$$

$$\begin{aligned}D_P'^{-}(o_{10}) &= \{o_1, o_{10}, o_{14}, o_{15}\}. \\ D_P'^{+}(o_{10}) &= \{o_4, o_7, o_8, o_{10}, o_{11}, o_{16}\}.\end{aligned}$$

According to Theorem 5–8, we have

$$\begin{aligned}\underline{P}\left(Cl_1^{\leqslant\prime}\right) &= \underline{P}\left(Cl_1^{\leqslant}\right) - M \cup N \\ &= \{o_1, o_2, o_{14}\} - \phi \cup \phi \\ &= \{o_1, o_2, o_{14}\}.\end{aligned}$$

Table 4 Deletion of one object.

Object	c_l	c_2	c_3	d
o_1	0.8	0.5	1.5	Cl_1
o_2	1	1.5	8	Cl_1
o_3	1	3	5	Cl_1
o_4	2.5	4	11	Cl_2
o_5	1.2	1	7	Cl_2
o_6	1.9	4.5	14	Cl_2
o_7	2.8	5.5	15	Cl_3
o_8	2.4	4	13	Cl_3
o_9	0.6	3	6	Cl_2
o_{11}	2.4	3.5	9	Cl_3
o_{12}	1.6	3	12	Cl_2
o_{13}	1.8	5	9.5	Cl_2
o_{14}	0.6	2	2.5	Cl_1
o_{15}	1	2	4.5	Cl_2
o_{16}	3	4.5	9	Cl_3

$$\begin{aligned}\overline{P}\left(Cl_1^{\leqslant\prime}\right) &= \overline{P}\left(Cl_1^{\leqslant}\right) - M \cup N \\ &= \{o_1, o_2, o_3, o_{10}, o_{14}, o_{15}\} - \{o_{10}\} \cup \phi \\ &= \{o_1, o_2, o_3, o_{14}, o_{15}\}.\end{aligned}$$

$$\begin{aligned}\underline{P}\left(Cl_2^{\leqslant\prime}\right) &= \underline{P}\left(Cl_2^{\leqslant}\right) - M \cup N \\ &= \underline{P}\left(Cl_2^{\leqslant}\right) - \{o_{10}\} \cup \phi \\ &= \{o_1, o_2, o_3, o_5, o_6, o_9, o_{12}, o_{13}, o_{14}, o_{15}\}.\end{aligned}$$

$$\begin{aligned}\overline{P}\left(Cl_2^{\leqslant\prime}\right) &= \overline{P}\left(Cl_2^{\leqslant}\right) - M \cup N \\ &= \overline{P}\left(Cl_2^{\leqslant}\right) - \{o_{10}\} \cup \phi \\ &= \{o_1, o_2, o_3, o_4, o_5, o_6, o_9, o_{11}, o_{12}, o_{13}, \\ &\quad o_{14}, o_{15}\}.\end{aligned}$$

$$\begin{aligned}\underline{P}\left(Cl_2^{\geqslant\prime}\right) &= \underline{P}\left(Cl_2^{\geqslant}\right) - M \cup N \\ &= \underline{P}\left(Cl_2^{\geqslant}\right) - \phi \cup \phi \\ &= \{o_4, o_5, o_6, o_7, o_8, o_9, o_{11}, o_{12}, o_{13}, o_{16}\}.\end{aligned}$$

$$\begin{aligned}\overline{P}\left(Cl_2^{\geqslant\prime}\right) &= \overline{P}\left(Cl_2^{\geqslant}\right) - M \cup N \\ &= \overline{P}\left(Cl_2^{\geqslant}\right) - \{o_{10}\} \cup \phi \\ &= \{o_3, o_4, o_5, o_6, o_7, o_8, o_9, o_{11}, o_{12}, o_{13}, \\ &\quad o_{15}, o_{16}\}.\end{aligned}$$

$$\begin{aligned}\underline{P}\left(Cl_3^{\geqslant\prime}\right) &= \underline{P}\left(Cl_3^{\geqslant}\right) - M \cup N \\ &= \{o_7, o_8, o_{16}\} - \phi \cup \phi \\ &= \{o_7, o_8, o_{16}\}.\end{aligned}$$

$$\begin{aligned}\overline{P}\left(Cl_3^{\geqslant}\right) &= \overline{P}\left(Cl_3^{\geqslant}\right) - M \cup N \\ &= \{o_4, o_7, o_8, o_{11}, o_{16}\} - \phi \cup \phi \\ &= \{o_4, o_7, o_8, o_{11}, o_{16}\}.\end{aligned}$$

6. INCREMENTAL UPDATE MATRIX ALGORITHM OF THE UNION SET'S APPROXIMATIONS AT INSERTION OR DELETION OF ONE OBJECT

According to the preceding methods using trisecting and acting in update of approximations at object's insertion or removal, their corresponding incremental update algorithms are proposed. All the algorithms are described by matrix considering that representation and calculation of matrix are intuitive and concise.

The Algorithm 1 is finished.

Assume that number of objects, number of attributes and number of dominance classes are n1, n2 and m, respectively. The time complexity of Algorithm 1 is equal to $\boldsymbol{O}(m \cdot (8 + 2 \cdot n2) \cdot n1)$. Thus it approximately equal to $\boldsymbol{O}(m \cdot n2 \cdot n1)$.

Algorithm 1: *A matrix-based incremental algorithm for updating approximations of unions of decision classes when a new object adds to an information system.*

Input:
(1) $\boldsymbol{D}_{\mathrm{P}}$ *at time t.*
(2) $\forall n \in T, \underline{P}\left(Cl_n^{\geqslant}\right), \overline{P}\left(Cl_n^{\geqslant}\right), \underline{P}\left(Cl_n^{\leqslant}\right), \overline{P}\left(Cl_n^{\leqslant}\right)$ *at time t.*
(3) An inserted object x^{+}.
Output:
$\forall n \in T, \underline{P}\left(Cl_n'^{\geqslant}\right), \overline{P}\left(Cl_n'^{\geqslant}\right), \underline{P}\left(Cl_n'^{\leqslant}\right), \overline{P}\left(Cl_n'^{\leqslant}\right)$.
Begin
Compute $\boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right)$ *and* $\boldsymbol{D}_{\boldsymbol{P}}^{-}\left(x^{+}\right)$.
Update $\boldsymbol{D}_{\boldsymbol{P}}$ *at time t to* $\boldsymbol{D}_{\boldsymbol{P}}'$ *at time t + 1.//insert* $\boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right)$ *and* $\boldsymbol{D}_{\boldsymbol{P}}^{-}\left(x^{+}\right)$ *//after the (i−1)th column and after the (i−1)th row.*
Update $Cl_n^{\geqslant}$ *and* $Cl_n^{\leqslant}$ *at time t to* $Cl_n'^{\geqslant}$ *and* $Cl_n'^{\leqslant}$ *at time t + 1.*
For n = *1,...,m do*
If (n < g) *Then// g is the index of the class which the object* x^{+} *is // assigned*
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right), \boldsymbol{Cl}_n^{\geqslant}\right]\right) == \left|\boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right)\right|$, *Then* $\underline{P}(\boldsymbol{Cl}_n^{\geqslant}) \leftarrow \underline{P}(\boldsymbol{Cl}_n^{\geqslant}) + \boldsymbol{x}^{+}$
End,
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}(:g), \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right)\right] == 0\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) + \boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right)$
End,
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}(:g), \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right)\right] != 0\right)$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) - \left[\boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right), \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right)\right]$ *End.*
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}(:g), \boldsymbol{Cl}_n^{\leqslant}\right] != 0\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) + \boldsymbol{x}^{+}$
End.
End.
If (n == g) *Then*
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right), \boldsymbol{Cl}_n^{\geqslant}\right]\right) == \left|\boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right)\right|$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) + \boldsymbol{x}^{+}$
End,
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}(:g), \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right)\right] == 0\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) + \boldsymbol{D}_{\boldsymbol{P}}^{+}\left(x^{+}\right)$
End.
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}(g:), \boldsymbol{Cl}_n^{\leqslant}\right] != \left|\boldsymbol{D}_{\boldsymbol{P}}(g:)\right|\right)$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) + \boldsymbol{x}^{+}$
End.
$\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) + \boldsymbol{D}_{\boldsymbol{P}}^{-}\left(x^{+}\right)$.
End.
End.
Return $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right), \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right), \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right)$ *and* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right)$ *at time t + 1.*
End.

The Algorithm 2 is finished.

Similar to Algorithm 1, the time complexity of Algorithm 2 is approximately equal to $\boldsymbol{O}(m \cdot n2 \cdot n1)$.

The non-incremental algorithm for computing the approximations is given below in order to compare its performance with that of the incremental algorithm in the next subsection.

Algorithm 2: *A matrix-based incremental algorithm for updating approximation of unions of decision classes when an object gets out of an information system.*

Input:
(1) $\boldsymbol{D}_{\boldsymbol{P}}$ at time t.
(2) $\forall n \in T, \underline{P}\left(Cl_n^{\geqslant}\right), \overline{P}\left(Cl_n^{\geqslant}\right), \underline{P}\left(Cl_n^{\leqslant}\right), \overline{P}\left(Cl_n^{\leqslant}\right)$ *at time t.*
(3) *An deleted object* x^{-}.
Begin
Update $\boldsymbol{D}_{\boldsymbol{P}}$ *at time t.//delete the i th column and the i th row*
Update $Cl_n^{\geqslant}$ *and* $Cl_n^{\leqslant}$ *at time t.*
For n = *1,...,* m *do*
If (n < g) *Then// g is the index of the class which the object* x^{-} *//is assigned.*
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{+}(x^{-}), \boldsymbol{Cl}_n^{\geqslant}\right] == \left|\boldsymbol{D}_{\boldsymbol{P}}^{+}(x^{-})\right|\right)$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) - \boldsymbol{x}^{-}$
End.
$\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) - \boldsymbol{x}^{-}$.
For each $\mathrm{u} \in \overline{P}\left(Cl_n^{\geqslant}\right) \cap D_P^{+}(x^{-})$
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{-}{}'(u), \boldsymbol{Cl}_n^{\geqslant}{}'\right] == 0\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) - \boldsymbol{u}$ *End.*
End
For each $\mathrm{u} \in D_P^{+}(x^{-}) \wedge \mathrm{u} \notin \underline{P}\left(Cl_n^{\leqslant}\right)$
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{-}{}'(u), \boldsymbol{Cl}_n^{\leqslant}{}'\right] = \left|D_P^{-}{}'(u)\right|\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) + \boldsymbol{u}$
End.
End
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{+}(x^{-}), \boldsymbol{Cl}_n^{\leqslant}\right] \neq 0\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) - \boldsymbol{x}^{-}$ *End.*
End.
If(*n* == *g*) *Then*
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{+}(x^{-}), \boldsymbol{Cl}_n^{\geqslant}\right] == \left|D_P^{+}(x^{-})\right|\right)$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) - \boldsymbol{x}^{-}$
End.
$\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \overline{P}\left(\boldsymbol{Cl}_n^{\geqslant}\right) - \boldsymbol{x}^{-}$
For each $\mathrm{u} \in \overline{P}\left(Cl_n^{\geqslant}\right) \cap D_P^{+}(x^{-})$
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{-}{}'(u), \boldsymbol{Cl}_n^{\geqslant}{}'\right) == 0\right]\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\geqslant}\right) - \boldsymbol{u}$ *End.*
End.
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}^{-}(x^{-}), \boldsymbol{Cl}_n^{\leqslant}\right] == \left|D_P^{-}(x^{-})\right|\right)$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) - \boldsymbol{x}^{-}$
End.
For each $\mathrm{u} \in \overline{P}\left(Cl_n^{\leqslant}\right) \cap D_P^{-}{}'(x^{-})$
If $\left(\left[\boldsymbol{D}_{\boldsymbol{P}}'^{+}(\mathrm{u}), \boldsymbol{Cl}_n'^{\leqslant}\right] == 0\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl}_n^{\leqslant}\right) - \boldsymbol{u}$ *End.*
End.
End.

If (n > g) *Then*
For each $u \notin \underline{P}\left(Cl_n^{\geqslant}\right) \wedge u \in D_p^-(x^-)$
If $\left(\left[\boldsymbol{D_P^{+\prime}}(u), \boldsymbol{Cl_n^{\geqslant\prime}}\right] == \left|D_P^{+\prime}(u)\right|\right)$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\geqslant}}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\geqslant}}\right) + \boldsymbol{u}$
End.
End.
If $\left(\left[\boldsymbol{D_P^-}(x^-), \boldsymbol{Cl_n^{\geqslant}}\right]\right) \neq 0$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\geqslant}}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\geqslant}}\right) - \boldsymbol{x}^-$ *End.*
If $\left(\left[\boldsymbol{D_P^-}(x^-), \boldsymbol{Cl_n^{\leqslant}}\right] == \left|D_P^-(x^-)\right|\right)$, *Then* $\underline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\leqslant}}\right) \leftarrow \underline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\leqslant}}\right) - \boldsymbol{x}^-$
End.
For each $u \in \overline{P}\left(Cl_n^{\leqslant}\right) \cap D_P^{-\prime}(x^-)$
If $\left(\left[\boldsymbol{D_P'^{+}}(u), \boldsymbol{Cl_n'^{\leqslant}}\right] == 0\right)$, *Then* $\overline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\leqslant}}\right) \leftarrow \overline{\boldsymbol{P}}\left(\boldsymbol{Cl_n^{\leqslant}}\right) - \boldsymbol{u}$ *End.*
End
End.
End.

The Algorithm 3 is finished.

The time complexity of Algorithm 3 is equal to $\boldsymbol{O}\left(m \cdot \left(n1^2 \cdot (n2+8) + 8 \cdot n1\right)\right)$. It is approximately equal to $\boldsymbol{O}\left(m \cdot n1^2 \cdot n2\right)$.

Algorithms 1–3 have the same space complexity due to the same memory space.

Algorithm 3: *A matrix-based algorithm for computing approximations of unions of decision classes.*

Input:
$IS = \left\{U, C \cup d, V, f\right\}, U/IND(d)$.
Output:
$\forall n \in T, \underline{P}\left(Cl_n^{\geqslant}\right), \overline{P}\left(Cl_n^{\geqslant}\right), \underline{P}\left(Cl_n^{\leqslant}\right), \overline{P}\left(Cl_n^{\leqslant}\right)$.
Begin
Computing $\boldsymbol{D}_p$, $Cl_n^{\geqslant}$, $Cl_n^{\leqslant}$ *by using IS.*
Computing matrix multiplications $\boldsymbol{D}_p \cdot \boldsymbol{Cl}_n^{\geqslant}$, $\boldsymbol{D}_p^T \cdot \boldsymbol{Cl}_n^{\geqslant}$, $\boldsymbol{D}_p \cdot \boldsymbol{Cl}_n^{\leqslant}$ *and* $\boldsymbol{D}_p^T \cdot \boldsymbol{Cl}_n^{\leqslant}$. *Denote by* $\boldsymbol{M}_1$, $\boldsymbol{M}_2$, $\boldsymbol{M}_3$ *and* $\boldsymbol{M}_4$ *respectively.*
Computing Column Vector by sum of row $sum\left(\boldsymbol{D}_n^T, 2\right)$ *and* $sum(D_n, 2)$.
$\underline{\boldsymbol{P}}\left(Cl_n^{\geqslant}\right) \leftarrow \left(\boldsymbol{M}_1 \geqslant sum\left(D_p^T, 2\right)\right)$.
$\overline{\boldsymbol{P}}\left(Cl_n^{\geqslant}\right) \leftarrow \left(\boldsymbol{M}_2 \geqslant ones(n, 1)\right)$.
$\underline{\boldsymbol{P}}\left(Cl_n^{\leqslant}\right) \leftarrow \left(\boldsymbol{M}_3 \geqslant sum\left(\boldsymbol{D}_p, 2\right)\right)$.
$\overline{\boldsymbol{P}}\left(Cl_n^{\leqslant}\right) \leftarrow \left(\boldsymbol{M}_4 \geqslant ones(n, 1)\right)$.
End.

7. THE TEST AND EVALUATION IN UCI DATA SET

Three data sets are selected (i.e., 'Wine' data set, 'Car evaluation' data set and 'Abalone' data set) from the machine learning data repository UCI (http://archive.ics.uci.edu/ml/) to test the performance of Algorithms 1 and 2 proposed in this paper and the existing Algorithm 3 (non-incremental updating algorithm) in order to verify the effectiveness of the proposed algorithms. Descriptions of the selected data sets are shown in Table 5.

Experimental Platform: CPU Intel Core i7-4510U (2.00 GHz), 8.0G Memory, Windows 8 operation system, Matlab7.0 development tool. Experimental method is as follows:

To show the time efficiency of dynamic algorithm (Algorithms 1 and 2) and compare with the existing static algorithm (Algorithm 3), each of selected data sets is divided into 10 sub-data sets. The generation of sub-data set follows four principles, take the 'wine' data set as an example to explain how to generate sub-data set from whole data set. Firstly, the 'wine' data set is divided into 10 subsets, each sub-data set is named as wine 1, wine 2, wine 3, wine 4, ... and wine 10, respectively. Secondly, the size of wine 1, wine 2, wine 3, ... and wine 9 is one-tenth, two-tenths, three-tenths, ... and nine-tenths of that of 'wine' data set, respectively, wine 10 is the copy of wine data set. Thirdly, there exists the following inclusion relation among these ten sub-data sets, the wine 10 contains wine 9, wine 9 contains wine 8, ..., wine 3 contains wine 2 and wine 2 contains wine1. Fourthly, the number of objects in a certain class of each sub-data set should be proportional to the size of the sub-data set in order to keep the original proportion of distribution of each class in whole data set. The distributions of each class in sub-data set of 'wine' are shown in Table 6.

The running time of the proposed Algorithms 1and 2 and the existing Algorithm 3 can be gained by executing the corresponding programs on each of these 10 sub-data sets. Concerned programs which update four approximations of DRSA while object set varies are all developed on Matlab7.0 platform.

The experimental results are depicted in Fig. 4, where the x-coordinate pertains to the test sub-data sets, while y-coordinate concerns the computing time of updating approximations. The following conclusions can be drawn from the experimental results on three UCI data sets:

i. The incremental updating matrix algorithms (Algorithms 1 and 2) have more effectiveness compared to the non-incremental updating matrix algorithm (Algorithm 3), showing that the time consume in incremental updating algorithm is obviously less that of the non-incremental ones.

ii. For the same data set, the computational times of the incremental updating matrix algorithms and the non-incremental updating matrix algorithm all grow up with the increasing size

Table 5 | A descriptions on the selected three data sets.

Name	Number of Attributes	Number of Objects	Number of Classes
Wine	13	178	3
Car evaluation	6	1728	4
Abalone	8	4177	29

Table 6 | The distribution of classes in each subset of Wine.

Sub-Data Set	Total Objects	Objects in Class 1	Objects in Class 2	Objects in Class 3
Wine 1	18	6	7	5
Wine 2	36	12	14	10
Wine 3	54	18	21	15
Wine 4	72	24	28	20
Wine 5	90	31	35	24
Wine 6	107	36	42	29
Wine 7	125	42	49	34
Wine 8	142	48	56	38
Wine 9	160	54	63	43
Wine 10	178	59	71	48

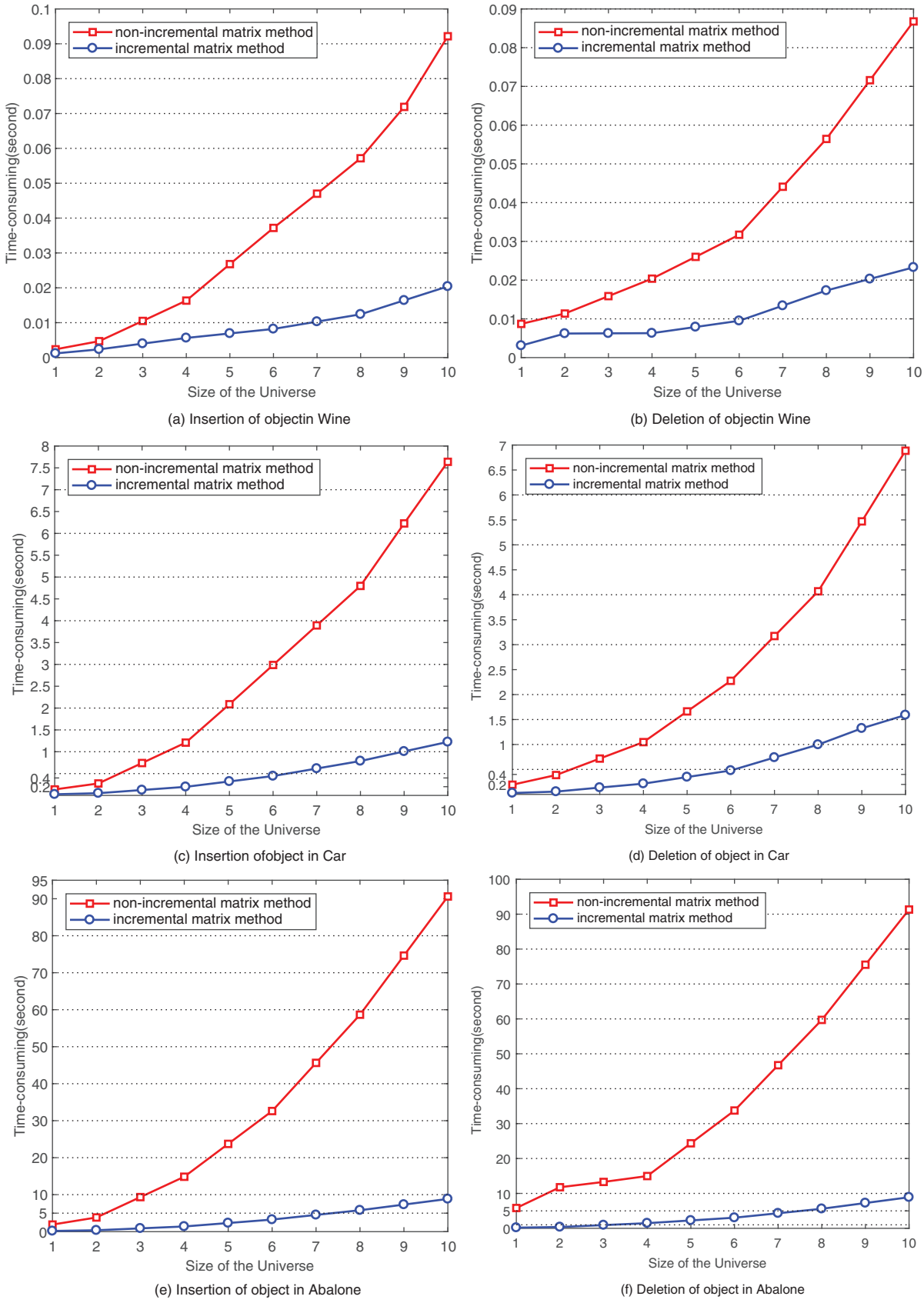

Figure 4 The curve of time-consuming for updating of approximations at insertion or deletion of a one objects.

of sub-data set. Furthermore, Algorithms 1 and 2 are much more faster than Algorithm 3, and the difference between incremental algorithms and the non-incremental algorithm are getting larger and larger while the size of sub-data set increases.

iii. The running times of Algorithm 3 increases sharply with the increasing size of data while the running times of Algorithms 1 and 2 increase very slowly, demonstrating that the larger the size of data, the greater the advantage of Algorithms 1 and 2.

iv. For the incremental update algorithm, the running time of object's insertion is more less than that of object's deletion in the same sub-data set. The reason is that there are more loop structure in Algorithm 2, which lead to more time consume.

It can be seen from the above four conclusions that the incremental update approaches of approximations with trisecting-and-acting model of three-way decisions are feasible and outperform the existing non-incremental approach.

8. CONCLUSIONS

Incrementally updating approximations in rough sets is a critical issue for knowledge update, maintenance and data mining related task in dynamic IS. In this paper, considering the dynamic scenario of one object's insertion or deletion, an incremental updated mechanism of approximations in DRSA is introduced from the perspective of three-way decisions and three update steps are derived from the updated mechanism. Then we suggested a concrete incremental updating approach of approximations in DRSA while the object set varies, in which Boolean column vectors are used as an expression tool of subsets as well as a calculation tool of approximations. The proposed methods can effectively update the four approximations of DRSA on the basis of the prior approximations. With a numerical example and some test results on UCI data set, we can conclude that the proposed incremental vector method for updating the approximations of DRSA is feasible and can effectively reduce the computational time in comparison to the existing non-incremental vector method when the set of objects changes. One of the further work is to investigate the approaches for updating approximations of DRSA under variations of attribute and attribute value and study the corresponding vector-based incremental algorithms.

CONFLICT OF INTEREST

The authors declare no conflicts of interest.

AUTHORS' CONTRIBUTIONS

All authors have contributed to this paper. The individual responsibilities and contribution of all authors can be described as follows: The idea of this paper was put forward by Lei Wang, he also wrote the paper. Min Li summarized the existing work of the research problem and presented the illustrative example in Section 5. The works in Section 3 were done by Jun Ye. The submission and revision of this paper was completed by Lei Wang and Xiang Yu. The tests on UCI data sets were performed by Ziqi Wang and Shaobo Deng. The other works of this paper were done by Lei Wang.

Funding Statement

This work was supported by the fund of Development Plan of Jiangxi Province of China for the Young and Middle-Aged Teachers in Universities to Pursue Study Abroad as a Visiting Scholar (Jiangxi Education Administration Letter [2016] No. 169), the National Natural Science Foundation of China (Nos. 61562061, 6136304 and 61703199), and the fund of Science and Technology Project of Ministry of Education of Jiangxi province of China (No. GJJ170995).

ACKNOWLEDGMENTS

The author would like to thank reviewers of the paper for their constructive comments which greatly improve the quality of this paper. This work was supported by the fund of Development Plan of Jiangxi Province of China for the Young and Middle-Aged Teachers in Universities to Pursue Study Abroad as a Visiting Scholar (Jiangxi Education Administration Letter [2016] No. 169), the National Natural Science Foundation of China (No. 61562061, No. 61363047, No. 61703199), and the fund of Science and Technology Project of Ministry of Education of Jiangxi province of China (No. GJJ170995).

REFERENCES

[1] Z. Pawlak, Rough sets, Int. J. Comp. Inf. Sci. 11 (1982), 341–356.
[2] Y.Y. Yao, Granular computing: basic issues and possible solutions, in Proceedings of 5th Joint Conference on Information Sciences, Atlantic City, New Jersey, 2000, pp. 186–189.
[3] Y.H. Qian, J.Y. Liang, W. Pedrycz, An efficient accelerator for attribute reduction from incomplete data in rough set framework, Pattern Recognit. 44 (2011), 1658–1670.
[4] X. Chen, W. Ziarko, Experiments with rough set approach to face recognition, Int. J. Intell. Syst. 26 (2011), 3–21.
[5] J. Blaszczynski, S. Greco, R. Slowinski, Inductive discovery of laws using monotonic rules, Eng. Appl. Artif. Intell. 25 (2012), 284–294.
[6] G.Q. Zhang, Z. Zheng, J. Lu, Q. He, An algorithm for solving rule sets-based bi-level decision problems, Comput. Intell. 27 (2011), 235–259.
[7] H. Li, X. Zhou, Risk decision making based on decision-theoretic rough set: a three-way view decision model, Int. J. Comput. Intell. Syst. 4 (2011), 1–11.
[8] H.S. Nguyen, Knowledge discovery by relation approximation: a rough set approach, Lect. Notes Comput. Sci. 4062 (2006), 103–106.
[9] A. Ganivada, S.K. Pal, A novel fuzzy rough granular neural network for classification, Int. J. Comput. Intell. Syst. 4 (2011), 1042–1051.
[10] S. Greco, B. Matarazzo, R. Slowinski, Rough approximation by dominance relations, Int. J. Intell. Syst. 17 (2002), 153–171.
[11] S. Greco, B. Matarazzo, R. Slowinski, Rough sets theory for multicriteria decision analysis, Eur. J. Oper. Res. 129 (2001), 1–47.
[12] S. Greco, B. Matarazzo, R. Slowinski, Rough approximation of a preference relation by dominance relations, Eur. J. Oper. Res. 117 (1999), 63–83.
[13] W. Pedrycz, R. Weber, Special issue on soft computing for dynamic data mining, Appl. Soft Comput. 8 (2008), 1281–1282.

[14] T.R. Li, D. Ruan, W. Geert, J. Song, Y. Xu, A rough sets based characteristic relation approach for dynamic attribute generalization in data mining, Knowl. Based Syst. 20 (2007), 485–494.
[15] J.B. Zhang, T.R. Li, D. Ruan, D. Liu, Neighborhood rough sets for dynamic data mining, Int. J. Intell. Syst. 27 (2012), 317–342.
[16] H.M. Chen, T.R. Li, D. Ruan, J.H. Lin, A rough-set based incremental approach for updating approximations under dynamic maintenance environments, IEEE Trans. Knowl. Data Eng. 25 (2013), 274–284.
[17] D. Liu, T.R. Li, J.B. Zhang, A rough set-based incremental approach for learning knowledge in dynamic incomplete information systems, Int. J. Approx. Reason. 55 (2014), 1764–1786.
[18] Y.Y. Zhang, T.R. Li, C. Luo, J.B. Zhang, H.M. Chen, Incremental updating of rough approximations in interval-valued information systems under attribute generalization, Inf. Sci. 373 (2016), 461–475.
[19] Y.Y. Huang, T.R. Li, C. Luo, H. Fujita, S.J. Horng, Dynamic variable precision rough set approach for probabilistic set-valued information systems, Knowl. Based Syst. 122 (2017), 131–147.
[20] A.P. Zeng, T.R. Li, D. Liu, J.B. Zhang, H.M. Chen, A fuzzy rough set approach for incremental feature selection on hybrid information systems, Fuzzy Sets Syst. 258 (2015), 39–60.
[21] C. Luo, T.R. Li, H.M. Chen, Dynamic maintenance of approximations in set-valued ordered decision systems under the attribute generalization, Inf. Sci. 257 (2014), 210–228.
[22] D. Liu, T.R. Li, J.B. Zhang, Incremental updating approximations in probabilistic rough sets under the variation of attributes, Knowl. Based Syst. 73 (2015), 81–96.
[23] L.N. Wang, X. Yang, Y. Chen, L. Liu, S.Y. An, P. Zhuo, Dynamic composite decision-theoretic rough set under the change of attributes, Int. J. Comput. Intell. Syst. 11 (2018), 355–370.
[24] H.M. Chen, T.R. Li, D. Ruan, Maintenance of approximations in incomplete ordered decision systems while attribute values coarsening or refining, Knowl. Based Syst. 31 (2012), 140–161.
[25] C. Luo, T.R. Li, H. Chen, H. Fujita, Y. Zhang. Incremental rough set approach for hierarchical multicriteria classification, Inf. Sci. 429 (2018), 72–87.
[26] H.M. Chen, T.R. Li, C. Luo, S.J. Horng, A rough set-based method for updating decision rules on attribute values'coarsening and refining, IEEE Trans. Knowl. Data Eng. 26 (2014), 2886–2899.
[27] S.Y. Li, T.R. Li, Incremental update of approximations in dominance-based rough sets approach under the variation of attribute values, Inf. Sci. 294 (2015), 348–361.
[28] X. Yang, T.R. Li, H. Fujita, D. Liu, Y.Y. Yao, A unified model of sequential three-way decisions and multilevel incremental processing, Knowl. Based Syst. 134 (2017), 172–188.
[29] H.M. Chen, T.R. Li, C. Luo, S.J. Horng, A decision-theoretic rough set approach for dynamic data mining, IEEE Trans. Fuzzy Syst. 23 (2015), 1–12.
[30] C. Luo, T.R. Li, Z. Yi, S.J. Horng, Matrix approach to decision-theoretic rough sets for evolving data, Knowl. Based Syst. 99 (2016), 123–134.
[31] Y.Y. Huang, T.R. Li, C. Luo, H. Fujita, S.J. Horng, Matrix-based dynamic updating rough fuzzy approximations for data mining, Knowl. Based Syst. 119 (2017), 273–283.
[32] S. Wang, T.R. Li, C. Luo, H.M. Chen, H. Fujita, Domain-wise approaches for updating approximations with multi-dimensional variation of ordered information systems, Inf. Sci. 478 (2019), 100–124.
[33] Y.G. Jing, T.R. Li, H. Fujita, Z. Yu, B. Wang, An incremental attribute reduction approach based on knowledge granularity with a multi-granulation view, Inf. Sci. 411 (2017), 23–38.
[34] Y.G. Jing, T.R. Li, H. Fujita, B.L. Wang, N. Cheng, An incremental attribute reduction method for dynamic data mining, Inf. Sci. 465 (2018), 202–218.
[35] C. Luo, T.R. Li, H.M. Chen, L.X. Lu, Fast algorithms for computing rough approximations in set-valued decision systems while updating criteria values, Inf. Sci. 299 (2015), 221–242.
[36] J.B. Zhang, T.R. Li, D. Ruan, D. Liu, Rough sets based matrix approaches with dynamic attribute variation in set-valued information system, Int. J. Approx. Reason. 53 (2012), 620–635.
[37] L. Wang, T.R. Li, Q. Liu, M. Li, A mtrix-based approach for maintenance of approximations under the variation of object set, J. Comput. Appl. Dev. 50 (2013), 1992–2004.
[38] Y.Y. Yao, An outline of a theory of three-way decisions, in Preceedings of RSCTC 2012, Chengdu, 2012, pp. 1–17.
[39] Y.Y. Yao, Three-way decisions and cognitive computing, Cogn. Comput. 8 (2016), 543–554.
[40] H. Y, C. Zhang, G.Y. Wang, A tree-based incremental overlapping clustering method using the three-way decision theory, Knowl. Based Syst. 91 (2016), 189–203.
[41] X. Yang, T.R. Li, H. Fujita, D. Liu, A sequential three-way approach to multi-class decision, Int. J. Approx. Reason. 104 (2019), 108–125.
[42] X. Yang, T.R. Li, D. Liu, H.M. Chen, C. Luo, A unified framework of dynamic three-way probabilistic rough sets, Inf. Sci. 420 (2017), 126–147.
[43] J.B. Liu, H.X. Li, X.Z. Zhou, B. Huang, T.X. Wang, An optimization-based formulation for three-way decisions, Inf. Sci. 495 (2019), 185–214.
[44] H. X. Li, L. B. Zhang, X. Z. Zhou, B. Huang, Cost-sensitive sequential three-way decision modeling using a deep neural network, Int. J. Approx. Reason. 85 (2017), 68–78.
[45] G.L. Liu, The Axiomatization of the rough set upper approximation operations, Fundam. Inform. 69 (2006), 331–342.
[46] P.J. Olver, O.C. Shakiban, Applied Linear Algebra, Springer International Publishing, Basel, Switzerland., 2018, pp. 129–182.
[47] W. Ziarko, Variable precision rough set model, J. Comput. Syst. Sci. 46 (1993), 39–59.