

Incremental Computing Approximations with the Dynamic Object set in Interval-valued Ordered Information System

Jianhang Yu, Weihua Xu^{*†}

School of Mathematics and Statistics

Chongqing University of Technology

Chongqing, 400054, P.R. China

yujh2013@foxmail.com, chxuwh@gmail.com

Abstract. Rough set theory has been successfully used in information system for classification analysis and knowledge discovery. The upper and lower approximations are fundamental concepts of this theory. The new information arrives continuously and redundant information may be produced with the time in real-world application. So, then incremental learning is an efficient technique for knowledge discovery in a dynamic database, which enables acquiring additional knowledge from new data without forgetting prior knowledge, which need to be updated incrementally while the object set get varies over time in the interval-valued ordered information system. In this paper, we analyzed the updating mechanisms for computing approximations with the variation of the object set. Two incremental algorithms respectively for adding and deleting objects with updating the approximations are proposed in interval-valued ordered information system. Furthermore, extensive experiments are carried out on six UCI data sets to verify the performance of these proposed algorithms. And the experiments results indicate the incremental approaches significantly outperform non-incremental approaches with a dramatic reduction in the computational time.

Keywords: Approximations; Dynamic database; Incremental learning; Interval-valued ordered information system

^{*} Address for correspondence: No. 69 Hongguang Road, Banan District, Chongqing, P.R. China

[†] Also works: Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information, Nanjing University of Science and Technology, Nanjing, 210094, P.R. China and Key Laboratory of Oceanographic Big Data Mining & Application of Zhejiang Province, Zhejiang Ocean University, Zhoushan, Zhejiang 316022, P R China

1. Introduction

Rough set theory was proposed by Pawlak[28-30], is an extension of the classical set theory and could be regarded as a mathematical and soft computing tool to handle imprecision, vagueness and uncertainty in data analysis. This relatively new soft computing methodology has received great attention in recent years, and its effectiveness has been confirmed successful applications in many science and engineering fields, such as pattern recognition, data mining, image processing, medical diagnosis and so on[18,19]. Rough set theory is built on the basis of the classification mechanism, it is classified as the equivalence relation in a specific universe, and the equivalence relation constitutes a partition of the universe. A concept or more precisely the extension of a concept is represented by a subset of a universe of objects and is approximated by a pair of definable concepts of a logic language. The main idea of rough set theory is the use of a known knowledge in knowledge base to approximate the inaccurate and uncertain knowledge. It seems to be fundamental importance to artificial intelligence and cognitive sciences. The classical rough set is through the indiscernibility relation obtained equivalence classes and construct the lower and upper approximations. In many application fields the preference-ordered relation play an important role. To solve this problem, Greco et al. have proposed an extension of Pawlak's rough set approach, which is called the Dominance-based rough set approach(DRSA)[11-15]. In DRSA, where condition attributes are criteria and classes and the dominance classes are sets of objects defined by using a dominance relation[4, 41].

In real-world application, data in information system are generated and collected dynamically, and the knowledge discovery by RST need to be updating accordingly[32]. The incremental technique is an effective method to updating knowledge by dealing with the new added-in data set without re-implementing the original data mining algorithm[26]. With respect to the different angles to recognize the dynamics in rough sets, there exist two main viewpoints. The first one is based on the view of information table. Since an information table consists of data objects, data attributes and data attribute values[25] recent researches focus on the three types of variations, namely, variation of objects[1,6,20-23,42], variation of attributes[24, 44], variation of attributes' values[5]. The second one is based on the view of pre-topology[31]. The classification of dynamics in rough sets is divided into two aspects: synchronic dynamics and diachronic dynamics[7]. Furthermore, Ciucci[9] listed four main streamlines to investigate dynamics in rough sets, namely, lower and upper approximations[3, 42], reduce and rules[8], quality indexes[16, 20] and formal logical[17, 27]. To sum up, both viewpoints provide a basic and clear framework on dynamic studies of rough sets. Shan and Ziarko presented a discernibility-matrix based incremental methodology to find all maximally generalized rules. Bang and Bien proposed another incremental inductive learning algorithm to find a minimal set of rules for a decision table without recomputing all the set of instances when another instance set is added into the universe [1]. Tong and An developed an algorithm based on the ϑ -decision matrix for incremental learning rules. They listed seven cases that would happen when a new sample enters the system. Zheng and Wang developed a rough set and rule tree based incremental knowledge acquisition algorithm, RRIA, to update knowledge more quickly when new objects are added or removed from a given dataset[44]. Hu et al. constructed a novel incremental attribute reduction algorithm when new objects are added into a decision information system. Błaszczyński and Słowiński discussed the incremental induction of decision rules from dominance-based rough approximations to select the most interesting representatives in the final set of rules. Fan et al. proposed an approach of incremental rule induction based on rough sets[10]. In addition, Liu et al. proposed an incremental approach as well as its algorithm for inducing interesting knowledge

when objects change over time [20]. Then, Liu et al. further introduced the incremental matrix and presented a new optimization approach for knowledge discovery[21]. Followed by Lius work, Li et al. proposed an incremental approach for updating approximations in dominance-based rough sets[23]. Zhang et al. proposed a method for dynamic data mining based on neighborhood rough sets[42], and they further presented a parallel method for computing rough set approximations [43]. As an efficient data analysis' technique, the rough set based incremental approaches have become one the hot topics on extraction of knowledge from changing data sets in recent decades and have achieved fruitful results.

However, mainly study on incremental computing approximations concerned in the certainly single-valued or set-valued information system, but little attention has been paid to the interval-valued information system and ordered information system. And they are very important type of data tables, and generalized models of single-valued information system. Xu and Qian et al. have did some studies in ordered information system[34-38]. In recent years, some problems of decision making have been investigated in the context of interval information system . Qian et al. introduced a dominance relation to interval information systems and interval decision tables and established a rough set approach based on dominance relation for decision-making analysis in the context of interval value[33]. Yang et al. investigated the interval-valued information system based on dominance relation[39, 40]. The upper and lower approximations are fundamental of studying in rough set theory. In this paper, we investigate the incremental approaches for updating approximations with dynamic object set in interval-valued ordered information system. We focus on updating approximations under the variation of the object set in interval-valued ordered information system. We proposed two incremental updating algorithms when the objectees are deleted or inserted, respectively. At last, the performances of two incremental algorithms are evaluated on serval variety of data sets.

The remainder of this paper is organized as follows. In Section 2, some basic concepts of RST and interval-valued ordered information systems are simply introduced. In Section 3, the principles and some illustrated examples for incremental updating approximations with the variation of object set are presented. We proposed the incremental algorithms for computing approximations based on the updating principles in Section 4. In Section 5, performance evaluations are illustrated and the experiment results have exhibited. The paper ends with conclusions shown in Section 6.

2. Rough sets and interval-valued ordered information system

In this section, Some basic concepts and results of rough sets are outlined and we introduce a dominance relation to an interval information system, the rough set model of interval information system and some of their important properties were introduced. More details can refer to Literature [11-15,28,29,33,40].

For a non-empty set U , we call it the universe of discourse. The class of all subsets of U is denoted by $P(U)$. For $X \in P(U)$, the equivalence relation R in a Pawlak approximation space (U, R) partitions the universe U into disjoint subsets. Such a partition of the universe is a quotient set of U and is denoted by $U/R = \{[x]_R | x \in U\}$, where $[x]_R = \{y \in U | (x, y) \in R\}$ is the equivalence class containing x with respect to R . In the view of granular computing, equivalence classes are the basic building blocks for the representation and approximation of any subset of the universe of discourse. Each equivalence class may be viewed as a granule consisting of indistinguishable elements. The basic concept $X \in P(U)$, one

can characterize X by a pair of upper and lower approximations which are

$$\begin{aligned}\overline{R}(X) &= \{x \in U \mid [x]_R \cap X \neq \emptyset\}, \\ \underline{R}(X) &= \{x \in U \mid [x]_R \subseteq X\}.\end{aligned}$$

Here, $pos(X) = \underline{R}(X)$, $neg(X) = \sim \overline{R}(X)$, $bn(X) = \overline{R}(X) - \underline{R}(X)$ are called the positive region, negative region, and boundary region of X , respectively.

Pawlak upper and lower approximations divided the universe into three disjoint regions, namely, positive region, negative region and boundary region. These regions have qualitative semantics and reflect the positive certainty, negative certainty and uncertainty, respectively. While the Pawlak rough set has a severe limitation. The relationship between equivalence classes and the basic set are strict that there are no fault tolerance mechanisms. Quantitative information about the degree of overlap of the equivalence classes and the basic set is not taken into consideration. Therefore, neither wider relationships nor quantitative information can be utilized. In fact, there are some degrees of inclusion relations between sets, and the extent of overlap of sets is important information to consider in applications. The classical rough set model must be improved and expansions of the model that include quantification are of particular value.

An interval-valued information system is a quadruple $I = (U, AT, V, f)$, where U is a finite non-empty set of objects and AT is a finite non-empty set of attributes, $v = \cup_{a \in AT} V_a$ and V_a is a domain of attribute a , $f : U \times AT \rightarrow V$ is a total function such that $f(x, a) \in V_a$ for every $a \in AT, x \in U$, called an information function, where V_a is a set of interval numbers. Denoted by

$$f(x, a) = [a^L(x), a^U(x)] = \{p \mid a^L(x) \leq p \leq a^U(x), a^L(x), a^U(x) \in \mathbf{R}\},$$

we call it the interval number x under the attribute a . In particular, $f(x, a)$ would degenerate into a real number if $a^L(x) = a^U(x)$. Under this consideration, we regard a single-valued information system as a special form of interval information system.

In particular decision-making analysis, we always consider a binary dominance relation between objects that are possibly dominant in terms of value of an attribute set in an interval information system. In general, an increasing preference and a decreasing preference are considered by a decision maker. If the domain of an attribute is ordered according to a decreasing or increasing preference, then the attribute is a criterion.

An interval-valued information system is called interval-valued ordered information system if all attributes are criteria, referred to as IvOIS. It is assumed that the domain of a criterion $a \in AT$ is completely pre-ordered by an outranking relation \geq_a and $x \geq_a y$ means that x is at least as good as y with respect to the criterion a . For a subset of attribute $A \subseteq AT$, we define $x \geq_A y \Leftrightarrow \forall a \in A, x \geq_a y$. In other words, x is at least as good as y with respect to all attributes in A . In the following, we introduce a dominance relation that identifies dominance classes to an interval-valued ordered information system. In a given IvOIS, we say that x dominates y with respect to $A \subseteq AT$ if $x \geq_A y$, and denoted by $xR_A^>y$. That is

$$R_A^> = \{(y, x) \in U \times U \mid y \geq_A x\}.$$

It means that if $(x, y) \in R_A^>$, then y dominates x with respect to A . In other words, y may have a better property than x with respect to A in reality. In the similar way, the relation $R_A^<$ (called a dominated relation) can be defined as follows:

$$R_A^< = \{(y, x) \in U \times U \mid x \geq_A y\}.$$

For $A \subset AT$ and $A = A_1 \cup A_2$, if the attributes set A_1 according to increasing preference and A_2 according to decreasing preference, then the two binary relations can be defined more precisely as follows:

$$\begin{aligned} R_A^{\geq} &= \{(y, x) \in U \times U | a_1^L(y) \geq a_1^L(x), a_1^U(y) \geq a_1^U(x), \forall a_1 \in A_1; \\ &\quad a_2^L(y) \leq a_2^L(x), a_2^U(y) \leq a_2^U(x), \forall a_2 \in A_2; \} \\ &= \{(y, x) \in U \times U | (y, x) \in R_A^{\geq}\}. \\ R_A^{\leq} &= \{(y, x) \in U \times U | a_1^L(y) \leq a_1^L(x), a_1^U(y) \leq a_1^U(x), \forall a_1 \in A_1; \\ &\quad a_2^L(y) \geq a_2^L(x), a_2^U(y) \geq a_2^U(x), \forall a_2 \in A_2; \} \\ &= \{(y, x) \in U \times U | (y, x) \in R_A^{\leq}\}. \end{aligned}$$

Let $I^{\geq} = (U, AT, V, f)$ be an interval-valued ordered information system and $A \subset AT$, from the above definition of R_A^{\geq} and R_A^{\leq} , the following properties can be easily obtained.

$$R_A^{\geq} = \bigcap_{a \in A} R_{\{a\}}^{\geq}, \quad R_A^{\leq} = \bigcap_{a \in A} R_{\{a\}}^{\leq}.$$

And R_A^{\geq}, R_A^{\leq} are reflexive, R_A^{\geq}, R_A^{\leq} are asymmetric and R_A^{\geq}, R_A^{\leq} are transitive.

The dominance class induced by the dominance relation R_A^{\geq} is the set of objects dominating x , i.e.

$$[x]_A^{\geq} = \{a_1^L(y) \geq a_1^L(x), a_1^U(y) \geq a_1^U(x) \forall a_1 \in A_1; a_2^L(y) \leq a_2^L(x), a_2^U(y) \leq a_2^U(x), \forall a_2 \in A_2\},$$

and the set of objects dominated by x as follows.

$$[x]_A^{\leq} = \{a_1^L(y) \leq a_1^L(x), a_1^U(y) \leq a_1^U(x) \forall a_1 \in A_1; a_2^L(y) \geq a_2^L(x), a_2^U(y) \geq a_2^U(x), \forall a_2 \in A_2\}.$$

Where $[x]_A^{\geq}$ describes the set of objects that may dominates x and $[x]_A^{\leq}$ describes the set of objects that may dominated by x in terms of A in an interval-valued ordered information system, which are called the A -dominating set and the A -dominated set with respect to $x \in U$, respectively.

In many real application regions, one also can define the dominance relation on the universe with interval values through using others, the more details can be found in reference. Furthermore, no matter which dominance relation can be obtained similar to any one what have been investigated. Therefore, we just only adopt the dominance relation R_A^{\geq} for studying interval-valued ordered information system in this paper. For simplicity and without any loss of generality, in the following we only consider attributes with increasing preference.

Example 2.1. An interval-valued ordered information system is presented in Table 1. It is a case of the diagnosis of myocardial infarction, where $U = \{x_1, x_2, \dots, x_{10}\}$ representatives of ten different patients and $AT = \{a_1, a_2, \dots, a_5\}$ representatives of several enzymes related to the diagnosis of myocardial infarction. Where a_1 represents aspartate amino transferase(AST), a_2 represents Lactate dehydrogenase (LDH) and isoenzyme, a_3 represents Alfa hydroxybutyrate dehydrogenase(α -HBDH), a_4 represents Creatine Kinase(CK), a_5 represents Creatine Kinase isoenzymes(CKMB). Compute the classification induced by the dominance relation R_{AT}^{\geq} . And the different decision attribute values mean different diagnosis results.

From the Table 1, using the above property can be obtained that

$$\begin{aligned} [x_1]_{AT}^{\geq} &= \{x_1, x_5, x_7, x_8\}, [x_2]_{AT}^{\geq} = \{x_2, x_7, x_9\}, [x_3]_{AT}^{\geq} = \{x_3, x_9\}, \\ [x_4]_{AT}^{\geq} &= \{x_1, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}, [x_5]_{AT}^{\geq} = \{x_5\}, [x_6]_{AT}^{\geq} = \{x_6, x_8, x_9\}, \\ [x_7]_{AT}^{\geq} &= \{x_7\}, [x_8]_{AT}^{\geq} = \{x_8\}, [x_9]_{AT}^{\geq} = \{x_9\}, [x_{10}]_{AT}^{\geq} = \{x_7, x_8, x_9, x_{10}\}. \end{aligned}$$

So we can find that dominance classes in U/R_{AT}^{\geq} do not constitute a partition of U in general, but constitute a covering of U .

Table 1. An interval-valued ordered information system.

U	AST	LDH	$\alpha - HBDH$	CK	$CKMB$	d
x_1	[10,40]	[100,240]	[105,195]	[5 ,195]	[0 ,24]	2
x_2	[10,30]	[80 ,210]	[80 ,180]	[10,190]	[0 ,24]	1
x_3	[12,45]	[105,248]	[100,210]	[7 ,203]	[0 ,23]	2
x_4	[5 ,30]	[60 ,80]	[90 ,160]	[0 ,180]	[0 ,10]	1
x_5	[10,46]	[110,246]	[105,195]	[6 ,198]	[0 ,26]	2
x_6	[10,30]	[90 ,200]	[96 ,206]	[5 ,195]	[3 ,24]	2
x_7	[13,60]	[100,240]	[115,200]	[20,260]	[5 ,30]	3
x_8	[10,50]	[120,260]	[115,210]	[8 , 196]	[5 ,28]	2
x_9	[16,80]	[140,260]	[102,300]	[40, 320]	[10,60]	3
x_{10}	[8 ,32]	[60 ,196]	[80 ,178]	[6 , 160]	[2 ,20]	1

Based on the above set, approximations will be considered with respect to a dominance relation R_A^{\geq} in an interval-valued ordered information system . The original rough set approach proved to be very useful in dealing with inconsistency problems following from the information granulation. The original rough set idea is failing, however ,when preference-orders of attributes domains are to be taken into account [2].

Let $I^{\geq} = (U, AT, V, f)$ be an interval-valued ordered information system. For any $X \subseteq U$ and $A \subseteq AT$, the lower and upper approximations of X with respect to a dominance relation R_A^{\geq} are defined as follows:

$$\begin{aligned} \underline{R}_A^{\geq}(X) &= \cup\{[x]_A^{\geq} \mid [x]_A^{\geq} \subseteq X\}, \\ \overline{R}_A^{\geq}(X) &= \cup\{[x]_A^{\geq} \mid [x]_A^{\geq} \cap X \neq \emptyset\}. \end{aligned}$$

From the definition, one can easily notice that $\underline{R}_A^{\geq}(X)$ is a set of objects that belong to X with certainty and $\overline{R}_A^{\geq}(X)$ is a set of objects that possibly belong to X . It is similarly to Pawlak rough set that the $Bn_A(X) = \overline{R}_A^{\geq}(X) - \underline{R}_A^{\geq}(X)$ denotes a boundary of the rough set. Moreover, one can easily obtain the following properties.

Let $I^{\geq} = (U, AT, V, f)$ be an interval-valued ordered information system. For $X, Y \subseteq U$, $A \subseteq AT$ and R_A^{\geq} a dominance relation, then following properties hold.

$$(1) \quad \underline{R}_A^{\geq}(\emptyset) = \overline{R}_A^{\geq}(\emptyset) = \emptyset, \underline{R}_A^{\geq}(U) = \overline{R}_A^{\geq}(U) = U;$$

- (2) $\underline{R}_A^{\geq}(X) \subseteq X \subseteq \overline{R}_A^{\geq}(X)$;
- (3) $\underline{R}_A^{\geq}(\underline{R}_A^{\geq}(X)) = \underline{R}_A^{\geq}(X), \overline{R}_A^{\geq}(\overline{R}_A^{\geq}(X)) = \overline{R}_A^{\geq}(X)$;
- (4) $\underline{R}_A^{\geq}(X) = \sim \underline{R}_A^{\geq}(\sim X), \overline{R}_A^{\geq}(X) = \sim \overline{R}_A^{\geq}(\sim X)$;
- (5) $\underline{R}_A^{\geq}(X) \subseteq \underline{R}_{AT}^{\geq}(X), \overline{R}_A^{\geq}(X) \supseteq \overline{R}_{AT}^{\geq}(X)$, and $Bn_{AT}(X) \subseteq Bn_A(X)$;
- (6) If $X \subseteq Y$, then $\underline{R}_A^{\geq}(X) \subseteq \underline{R}_A^{\geq}(Y), \overline{R}_A^{\geq}(X) \subseteq \overline{R}_A^{\geq}(Y)$;
- (7) $\underline{R}_A^{\geq}(X \cap Y) = \underline{R}_A^{\geq}(X) \cap \underline{R}_A^{\geq}(Y), \overline{R}_A^{\geq}(X \cup Y) = \overline{R}_A^{\geq}(X) \cup \overline{R}_A^{\geq}(Y)$;
- (8) $\underline{R}_A^{\geq}(X \cup Y) \supseteq \underline{R}_A^{\geq}(X) \cup \underline{R}_A^{\geq}(Y), \overline{R}_A^{\geq}(X \cap Y) \subseteq \overline{R}_A^{\geq}(X) \cap \overline{R}_A^{\geq}(Y)$.

The lower and upper approximations of X with respect to the dominance relation $R_A \geq$ can be used to extract dominance rules by a decision maker, where one can extract dominance rules with certainty by using \underline{R}_A^{\geq} and can extract possible dominance rules by using $Bn_A(X) = \overline{R}_A^{\geq}(X) - \underline{R}_A^{\geq}(X)$.

Example 2.2. (Continued from Example 2.1). Consider the interval-valued ordered information system as Table 1. Let $A = \{a_1, a_2, \dots, a_5\} = AT$ and an object set $X = D_2 = \{x_1, x_3, x_5, x_6, x_8\}$, compute the rough sets of D_2 approximated by U/R_{AT}^{\geq} . According to the definition of approximations and Example 2.1, the rough set $\underline{R}_A^{\geq}(D_2)$ and $\overline{R}_A^{\geq}(D_2)$ can be obtained as follows.

$$\underline{R}_A^{\geq}(D_2) = \{x_5, x_8\}, \quad \overline{R}_A^{\geq}(D_2) = \{x_1, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}.$$

So the boundary of the rough set is $Bn_A(D_2) = \{x_1, x_3, x_4, x_6, x_7, x_9\}$.

3. Theories for incremental updating approximations in an IvOIS under the variation of objects

With the dynamic object set of interval-valued ordered information system, the structure of information granules in the information system may over time which leads to the change of knowledge induced by RST. In medical diagnosis the data does not usually remain a stable condition. Some objects will be deleted from the original information system with the patient’s cure or changing hospital and some objects will be inserted into the original information system with new patients arrive. So it is in other areas of science. Then the discovered knowledge may become invalid or some new implicit information may emerge in the whole updated information system. Rather than restarting from scratch by non-incremental or batch learning algorithm for each update, developing an efficient incremental algorithm to avoid unnecessary computations by utilizing the previous data structures or results are thus desired.

In this section, we investigate the variation of approximations of the dynamic IvOIS when the object set evolves over time while the attribute set remains constant. We assume the process for incremental update the approximations lasts two stages, namely, from time t to time $t + 1$. By considering the objects may enter into or get out of an information system at time $t + 1$ and we denote a dynamic IvOIS at time t as $I^{\geq} = (U, AT \cup \{d\}, V, f)$, and at time $t + 1$ the original information system change into $(I^{\geq})' = (U', AT \cup \{d\}, V', f)$ after insertion or deletion of objects. And we denote the decision classes and the A – dominating set as D_i and $[x]_A^{\geq}$, respectively at time t , which are denoted as D_i' and $([x]_A^{\geq})'$ respectively at time $t + 1$. The lower and upper approximations of decision class D_i with

respect to $A \subseteq AT$ are denoted as $\underline{R}_A^{\geq}(D_i)$ and $\overline{R}_A^{\geq}(D_i)$, respectively at time t , which are denoted as $\underline{R}_A^{\geq}(D_i)'$ and $\overline{R}_A^{\geq}(D_i)'$, respectively at time $t+1$. Here, we only investigate the incremental approach for updating approximations in the cases that a single object enter and get out of the interval-valued ordered information system. The change of multiple objects can be seen as the cumulative change of a single object. The approximations can be updated step by step through the updating principles in the case that a single object varies.

3.1. Theories for Incremental Computing Approximations with the Deletion of an Object in an IvOIS

Given an IvOIS $I^{\geq} = (U, AT \cup \{d\}, V, f)$ at time t , the deletion of object $x^- \in U$ will change the original information system and information granules $[x]_A^{\geq}(x \in U \text{ and } A \subseteq AT)$ and the equivalence decision classes $D_i(i \in \{1, \dots, r\})$. The approximations of D_i will change accordingly. Here, we investigate the principles for updating approximations of D_i as two cases:(1) The deleted object x^- belongs to D_i . (2)The deleted object x^- does not belongs to D_i .

Case 1. The deleted object x^- belongs to D_i , namely, $x^- \in D_i$.

Proposition 3.1. Let $I^{\geq} = (U, AT \cup \{d\}, V, f)$ be an IvOIS and any $A \subseteq AT$. When $x^- \in D_i(i \in \{1, \dots, r\})$ is deleted from U , we have the following properties about $\underline{R}_A^{\geq}(D_i)'$ and $\overline{R}_A^{\geq}(D_i)'$.

- (1) If $x^- \in \underline{R}_A^{\geq}(D_i)$, then $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) - \{x^-\}$. Otherwise $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$.
- (2) $\overline{R}_A^{\geq}(D_i)' = (\overline{R}_A^{\geq}(D_i) - [x^-]_A^{\geq}) \cup \Delta_1^-$, where $\Delta_1^- = \{x | x \in [x^-]_A^{\geq} \cap \Delta_2^-\}$ and $\Delta_2^- = \cup_{x \in D_i - \{x^-\}} [x]_A^{\geq}$.

Proof:

(1) If $x^- \in D_i$ is deleted from the universe U , we have $U' = U - \{x^-\}$ and $D_i' = D_i - \{x^-\}$. So for any $x \in U'$, we have $([x]_A^{\geq})' = [x]_A^{\geq} - \{x^-\}$. If $[x]_A^{\geq} \subseteq D_i$, then $([x]_A^{\geq})' \subseteq D_i'$. It's similar that if $[x]_A^{\geq}$ not $\subseteq D_i$ then $([x]_A^{\geq})'$ not $\subseteq D_i'$. So, from the definition of lower and upper approximations, we can get that for any $x \in U'$, if $x \in \underline{R}_A^{\geq}(D_i)$, then $x \in \underline{R}_A^{\geq}(D_i)'$ and if $x \notin \underline{R}_A^{\geq}(D_i)$ then $x \notin \underline{R}_A^{\geq}(D_i)'$. Hence, it is easy to obtain if $x^- \in \underline{R}_A^{\geq}(D_i)$, then $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) - \{x^-\}$. Otherwise, the lower approximation of D_i should be remain constant, *i.e.* $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$.

(2)According to the definition, we have the $\overline{R}_A^{\geq}(D_i) = \cup\{[x]_A^{\geq} | [x]_A^{\geq} \cap D_i \neq \emptyset\}$. Thus when the object $x^- \in D_i$ is deleted from U , the A -dominating set $[x^-]_A^{\geq}$ should be removed from the upper approximation $\overline{R}_A^{\geq}(D_i)$. It's mean $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i) - [x^-]_A^{\geq}$. However, it may be exist $x \in D_i - \{x^-\}$ satisfies that $\Delta_1^- = [x]_A^{\geq} \cap [x^-]_A^{\geq} \neq \emptyset$ and the object which $x \in [x^-]_A^{\geq}$, where $x' \in (D_i - \{x^-\})$ should not be removed from $\overline{R}_A^{\geq}(D_i)$. Therefore, we can obtain $\overline{R}_A^{\geq}(D_i)' = (\overline{R}_A^{\geq}(D_i) - [x^-]_A^{\geq}) \cup \Delta_1^-$, where $\Delta_1^- = \{x | x \in [x^-]_A^{\geq} \cap \Delta_2^-\}$ and $\Delta_2^- = \cup_{x \in D_i - \{x^-\}} [x]_A^{\geq}$.

Thus, the Proposition 3.1 is proved. \square

Example 3.1. (Continued from Example2.2). For Table 1, according to Proposition 3.1, we compute the lower approximation of D_2 by deleting x_3 and x_5 , the upper approximation by deleting x_6 from U , where $D_2 = \{x_1, x_3, x_5, x_6, x_8\}$, respectively.

(1) Assume the object x_3 be deleted from Table 1, so $U' = U - \{x_3\}$. We can find the $x_3 \in D_2$ but $x_3 \notin \underline{R}_A^{\geq}(D_2)$. Therefore, $\underline{R}_A^{\geq}(D_2)' = \{x_5, x_8\}$. Let the object x_5 be deleted from Table 1, so $U' = U - \{x_5\}$. We have $x_5 \in D_2$ and $x_5 \in \underline{R}_A^{\geq}(D_2)$. Therefore, $\underline{R}_A^{\geq}(D_2)' = \underline{R}_A^{\geq}(D_2) - \{x_5\} = \{x_8\}$

(2) Let the object x_6 be deleted from Table 1, so $U' = U - \{x_6\}$. We have $x_6 \in D_2$, $\Delta_2^- = \{x_1, x_3, x_5, x_7, x_8, x_9\}$. Then $\Delta_1^- = \{x_8, x_9\}$ and $\underline{R}_A^{\geq}(D_2)' = (\underline{R}_A^{\geq}(D_2) - [x_6]_A^{\geq}) \cup \Delta_1^- = \{x_1, x_3, x_4, x_5, x_7, x_8, x_9\}$.

Case2. The deleted object x^- does not belongs to D_i , namely $x^- \notin D_i$.

Proposition 3.2. Let $I^{\geq} = (U, AT \cup \{d\}, V, f)$ be an IvOIS and any $A \subseteq AT$. When $x^- \notin D_i (i \in \{1, \dots, r\})$ be deleted from U , we have the following properties about $\underline{R}_A^{\geq}(D_i)'$ and $\overline{R}_A^{\geq}(D_i)'$.

(1) $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) \cup \Delta_1^-$. Where $\Delta_1^- = \{x | x \in (D_i - \underline{R}_A^{\geq}(D_i)), ([x]_A^{\geq})' \subseteq D\}$, if $x^- \in [x]_A^{\geq}$ then $([x]_A^{\geq})' = [x]_A^{\geq} - \{x^-\}$ otherwise $([x]_A^{\geq})' = [x]_A^{\geq}$.

(2) If $x^- \in \underline{R}_A^{\geq}(D_i)$ then $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) - \{x^-\}$. Otherwise $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i)$.

Proof:

(1) Based on definition, we have for any $x \in D_i$, if $x \in \underline{R}_A^{\geq}(D_i)$ then $[x]_A^{\geq} \subseteq D_i$. When the object $x^- \notin D_i$ is deleted from the universe U , we have that $U' = U - \{x^-\}$ and $D_i' = D_i$. So, $\forall x \in U'$, $([x]_A^{\geq})' = [x]_A^{\geq} - \{x^-\}$. It is easy to get that if $[x]_A^{\geq} \subseteq D_i$ then $([x]_A^{\geq})' \subseteq D_i'$. Thus, for $\forall x \in \underline{R}_A^{\geq}(D_i) \Rightarrow x \in \underline{R}_A^{\geq}(D_i)'$. On the other hand, for $\forall x \in D_i - \underline{R}_A^{\geq}(D_i)$, we can get $[x]_A^{\geq} \subseteq D_i$. However, it may exist $x^- \in [x]_A^{\geq}$ such that $([x]_A^{\geq})' \subseteq D_i$ after the deletion of x^- . Then the x should be added to $\underline{R}_A^{\geq}(D_i)'$, namely, $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) \cup \{x\}$. Therefore, we have $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) \cup \Delta_1^-$, where $\Delta_1^- = \{x | x \in D_i - \underline{R}_A^{\geq}(D_i), ([x]_A^{\geq})' \subseteq D_i\}$, $([x]_A^{\geq})' = [x]_A^{\geq} - \{x^-\}$.

(2) According to definition, we have $\overline{R}_A^{\geq}(D_i) = \cup\{[x]_A^{\geq} | [x]_A^{\geq} \cap D_i \neq \emptyset\}$. Since the deleted object $x^- \notin D_i$, there exists an object $x \in D_i$ stratifies $x^- \in [x]_A^{\geq}$, if $x^- \in \overline{R}_A^{\geq}(D_i)$. Therefore, when the object x^- is deleted, we can get $([x]_A^{\geq})' = [x]_A^{\geq} - \{x^-\}$. Then $\overline{R}_A^{\geq}(D_i)' = \cup_{x \in D_i} ([x]_A^{\geq})' = \overline{R}_A^{\geq}(D_i) - \{x^-\}$. On the other hand, if $x^- \notin \overline{R}_A^{\geq}(D_i)$, we have $\forall x \in D_i$, $x^- \notin [x]_A^{\geq}$. Hence, the upper approximation of D_i will remain constant, namely, $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i)$.

Thus, the proof is fulfilled. \square

Example 3.2. (Continued from Example 2.2). For Table 1, according to Proposition 3.2, we compute the lower approximation of D_2 by deleting x_{10} from U , the upper approximation of D_2 by deleting x_9 and x_{10} from U .

(1) Assume the object x_{10} be deleted from Table 1, so $U' = U - \{x_{10}\}$. We have $x_{10} \notin D_2$, $D_2 - \underline{R}_A^{\geq}(D_2) = \{x_1, x_3, x_6\}$, and $([x_1]_A^{\geq})'$, $([x_3]_A^{\geq})'$, $([x_6]_A^{\geq})'$ not $\subseteq D_2$. Therefore, $\Delta_1^- = \emptyset$ and $\underline{R}_A^{\geq}(D_2)' = \underline{R}_A^{\geq}(D_2) \cup \Delta_1^- = \{x_5, x_8\}$.

(2) If the object x_9 be deleted from Table 1, then $U' = U - \{x_9\}$. We have $x_9 \notin D_2$ but $x_9 \in \overline{R}_A^{\geq}(D_2)$. Therefore, $\overline{R}_A^{\geq}(D_2)' = \overline{R}_A^{\geq}(D_2) - \{x_9\} = \{x_1, x_3, x_4, x_5, x_6, x_7, x_8\}$. Let the object x_{10} be deleted from Table 1, so $U' = U - \{x_{10}\}$. We have $x_{10} \notin D_2$ and $x_{10} \notin \overline{R}_A^{\geq}(D_2)$. Therefore, $\overline{R}_A^{\geq}(D_2)' = \overline{R}_A^{\geq}(D_2) = \{x_1, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$.

3.2. Theories for Incremental Computing Approximations with the Insertion of a New Object in an IvOIS

Given an interval-valued ordered information system(IvOIS) $I^{\geq} = (U, AT \cup \{d\}, V, f)$ at time t , when the information system is updated by inserting a new object x^+ into the universe U at time $t + 1$, where denotes the inserted object as x^+ . There are two situations may occur: (1) x^+ forms a new decision class, namely, for any $x \in U$, $f(x, d) \neq f(x^+, d)$; (2) x^+ does not forms a new decision class, namely, exist $x \in U$, $f(x, d) = f(x^+, d)$. The different between the two situation is, in the first situation, in addition to updating the approximations of the equivalence classes, we need to compute the approximations for the new decision class. Firstly, for updating the approximations of the equivalence classes D_i where $i \in \{1, \dots, r\}$ when inserting an object x^+ , we investigate the principles through two cases similar to the approach taken in the model of deletion: (1) The inserted object x^+ will belong to D_i , it's mean $f(x, d) = f(x^+, d)$, where $x \in D_i$; (2) The inserted object x^+ will not belong to D_i , namely, $f(x, d) \neq f(x^+, d)$, for any $x \in D_i, i \in \{1, \dots, r\}$.

To illustrate our incremental methods for updating approximations when inserting a new object x^+ into interval-valued ordered information system(IvOIS). We assume that the objects in Table 2 will be inserted into Table 1, the Table 2 are given as follows.

Table 2. The object inserted into the interval information system

U	AST	LDH	$\alpha - HBDH$	CK	$CKMB$	d
x_{11}	[10, 50]	[110, 250]	[115, 210]	[8, 195]	[5, 27]	2
x_{12}	[12, 45]	[105, 248]	[100, 210]	[7, 203]	[0, 23]	2
x_{13}	[10, 30]	[90, 200]	[96, 206]	[5, 195]	[3, 24]	2
x_{14}	[5, 30]	[60, 80]	[90, 160]	[0, 180]	[0, 10]	1
x_{15}	[30, 100]	[200, 600]	[100, 600]	[40, 800]	[10, 60]	4

Case 1. The inserted object x^+ will belong to D_i , namely $x^+ \in D_i, i \in \{1, \dots, r\}$.

Proposition 3.3. Let $I^{\geq} = (U, AT \cup \{d\}, V, f)$ be an IvOIS and any $A \subseteq AT$. When the object $x^+ \in D_i (i \in \{1, \dots, r\})$ be inserted into U , we have the following properties about $\underline{R}_A^{\geq}(D_i)'$ and $\overline{R}_A^{\geq}(D_i)'$.

(1) If $[x^+]_A^{\geq} \subseteq D_i'$, where $D_i' = D_i \cup \{x^+\}$ then $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) \cup \{x^+\}$. Otherwise, $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$.

(2) $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i) \cup [x^+]_A^{\geq}$.

Proof:

(1) According to definition, we have for any $x \in D_i$, if $[x]_A^{\geq} \subseteq D_i$, then $x \in \underline{R}_A^{\geq}(D_i)$. Thus, when the object x^+ is inserted into U , we have $D_i' = D_i \cup \{x^+\}$. For any $x \in D_i$, if $x^+ \in [x]_A^{\geq}$ then $([x]_A^{\geq})' = [x]_A^{\geq} \cup \{x^+\}$. That is, if $[x]_A^{\geq} \subseteq D_i$ then $([x]_A^{\geq})' \subseteq D_i'$. If $[x]_A^{\geq}$ not $\subseteq D_i$, then $([x]_A^{\geq})'$ not $\subseteq D_i'$. It follows that if $x \in \underline{R}_A^{\geq}(D_i)$, then $x \in \underline{R}_A^{\geq}(D_i)'$. If $x \notin \underline{R}_A^{\geq}(D_i)$, then $x \notin \underline{R}_A^{\geq}(D_i)'$. Therefore, if $[x]_A^{\geq} \subseteq D_i'$, we have $x^+ \in \underline{R}_A^{\geq}(D_i)'$ and $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) \cup \{x^+\}$. Otherwise, $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$.

(2) When the object x^+ be inserted into U that $U' = U \cup \{x^+\}$. According to definition, we have $\overline{R_A^{\geq}}(D_i)' = \cup_{x \in D_i'} ([x]_A^{\geq})'$. Since $D_i' = D_i \cup \{x^+\}$, then we have $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i)' \cup [x^+]_A^{\geq}$. Because for $\forall x \in U$ there $([x]_A^{\geq})' = [x]_A^{\geq} \cup \{x^+\}$ or $([x]_A^{\geq})' = [x]_A^{\geq}$ and the object $x^+ \in [x^+]_A^{\geq}$, we can obtain that $\overline{R_A^{\geq}}(D_i)' = \cup_{x \in D_{i+1}} [x]_A^{\geq} \cup [x^+]_A^{\geq} = \overline{R_A^{\geq}}(D_i) \cup [x^+]_A^{\geq}$.

Thus, the proof is finished. \square

Example 3.3. (Continued from Example2.2). For Table 1, according to Proposition 3.3, we compute the lower approximation of D_2 when the objects x_{11} and x_{12} are inserted, the upper approximation of D_2 when the object X_{13} in Table 2 inserted into the universe U , respectively.

(1) Let the object x_{11} in Table 2 be inserted into Table 1, so the $U' = U \cup \{x_{11}\}$. Since $f(x_{11}, d) = 2$ then $D_2' = D_2 \cup \{x_{11}\}$. Because of $[x_{11}]_A^{\geq} = \{x_8, x_{11}\} \subseteq D_2'$, we have $\overline{R_A^{\geq}}(D_2)' = \overline{R_A^{\geq}}(D_2) \cup \{x_{11}\} = \{x_5, x_8, x_{11}\}$. Assume the object x_{12} in Table 2 be inserted into Table 1 and $U' = U \cup \{x_{12}\}$. Since $f(x_{12}, d) = 2$ then $D_2' = D_2 \cup \{x_{12}\}$. Because of $[x_{12}]_A^{\geq} = \{x_3, x_9, x_{12}, x_{15}\}$ not $\subseteq D_2'$ then we have $\overline{R_A^{\geq}}(D_2)' = \overline{R_A^{\geq}}(D_2) = \{x_5, x_8\}$.

(2) Assume the object x_{13} in Table 2 be inserted into Table 1 and $U' = U \cup \{x_{13}\}$. Since $f(x_{13}, d) = 2$ then $D_2' = D_2 \cup \{x_{13}\}$ and $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i) \cup [x_{13}]_A^{\geq} = \{x_1, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{13}\}$.

Case 2. The inserted object x^+ will not belong to D_i , namely, $x^+ \notin D_i, i \in \{1, \dots, r\}$.

Proposition 3.4. Let $I^{\geq} = (U, AT \cup \{d\}, V, f)$ be an IvOIS and any $A \subseteq AT$. When the object $x^+ \notin D_i (i \in \{1, \dots, r\})$ be inserted into U , we have the following properties about $\overline{R_A^{\geq}}(D_i)'$ and $\overline{R_A^{\geq}}(D_i)'$.

(1) $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i) \cup \Delta_1^+$, where $\Delta_1^+ = \{x | x \in \overline{R_A^{\geq}}(D_i), x^+ \in ([x]_A^{\geq})'\}$.

(2) If there exists $x \in D_i$ such that $x^+ \in [x]_A^{\geq}$, then $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i) \cup \{x^+\}$. Otherwise $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i)$.

Proof:

(1) When the object x^+ be inserted into U , since $f(x, d) = f(x^+, d) (x \in D_i)$ we have $U' = U \cup \{x^+\}$ and $D_i' = D_i$. For $\forall x \in D_i'$ there $([x]_A^{\geq})' = [x]_A^{\geq}$ or $([x]_A^{\geq})' = [x]_A^{\geq} \cup \{x^+\}$. We have if $[x]_A^{\geq}$ not $\subseteq D_i$ then $[x]_A^{\geq}$ not $\subseteq D_i'$. That is, if $x \notin \overline{R_A^{\geq}}(D_i)$ then $x \notin \overline{R_A^{\geq}}(D_i)'$. Hence, we only consider the object $x^+ \in \overline{R_A^{\geq}}(D_i)$, namely, $D_i \subseteq [x]_A^{\geq}$. When the object x^+ be inserted into universe U , there may exist that $([x]_A^{\geq})' = [x]_A^{\geq} \cup \{x^+\}$ then $([x]_A^{\geq})'$ is not included by $D_i' = D_i$, namely, $x \notin \overline{R_A^{\geq}}(D_i)'$. Therefore, we have $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i) - \Delta_1^+$, where $\Delta_1^+ = \{x | x \in \overline{R_A^{\geq}}(D_i), x^+ \in ([x]_A^{\geq})'\}$.

(2) When the object x^+ is inserted into U , since $f(x, d) \neq f(x^+, d)$, we can obtain $U' = U \cup \{x^+\}$ and $D_i' = D_i$. Then, for $\forall x \in D_i'$, if $x^+ \in [x]_A^{\geq}$ then $([x]_A^{\geq})' = [x]_A^{\geq} \cup \{x^+\}$. And we have $x^+ \in \overline{R_A^{\geq}}(D_i)'$, that is, $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i) \cup \{x^+\}$. Otherwise, if for $\forall x \in D_i, x^+ \notin ([x]_A^{\geq})'$, that is, $([x]_A^{\geq})' = [x]_A^{\geq}$. Then we can get $\overline{R_A^{\geq}}(D_i)' = \overline{R_A^{\geq}}(D_i)$.

Thus, the proof is accomplished. \square

Example 3.4. (Continued from Example2.2). For Table 1, according to Proposition 3.4, we compute the lower and upper approximations of D_2 when the object x_{14} in Table 2 into the universe U .

(1) Assume the object x_{14} in Table 2 insert into Table 1, and $U' = U \cup \{x_{14}\}$. Since $f(x_{14}, d) = 1$ then $D'_2 = D_2$ remain unchanged. Because of $\underline{R}_A^{\geq}(D_2) = \{x_5, x_8\}$, and $x_{14} \notin ([x_5]_A^{\geq})', ([x_8]_A^{\geq})'$. Hence, we have $\Delta_1^+ = \emptyset$ then $\underline{R}_A^{\geq}(D_2)' = \underline{R}_A^{\geq}(D_2) = \{x_5, x_8\}$.

(2) Let the object x_{14} in Table 2 insert into Table 1 and $U' = U \cup \{x_{14}\}$. Since $f(x_{14}, d) = 1$, then $D'_2 = D_2$ remain unchanged. Because of $\underline{R}_A^{\geq}(D_2) = \{x_1, x_3, x_5, x_6, x_7, x_8, x_9\}$, and $x_{15} \in ([x_2]_A^{\geq})'$, that is $x_{15} \in ([x_2]_A^{\geq})'$. Hence, we have $\underline{R}_A^{\geq}(D_2)' = \underline{R}_A^{\geq}(D_2) \cup \{x_{15}\} = \{x_1, x_3, x_5, x_6, x_7, x_8, x_9, x_{15}\}$.

Based on above investigate, we can compute the lower and upper approximations of the existed equivalence decision classes D_i where $i \in \{1, \dots, r\}$ when inserting a new object into IvOIS. However, when a new object x^+ is inserted into the universe U , it might happen that x^+ will generate a new decision class, namely, $\forall x \in U, f(x, d) \neq f(x^+, d)$. Then the universe $U' = U \cup \{x^+\}$ will be divided into $r + 1$ partitions and $D_{r+1} = \{x^+\}$. At this point, in addition to updating the approximations of new decision class D_{r+1} .

Proposition 3.5. Let $I^{\geq} = (U, AT \cup \{d\}, V, f)$ be an IvOIS and any $A \subseteq AT$. When the object x^+ be inserted into U , if for $\forall x \in U, f(x, d) \neq f(x^+, d)$, then the lower approximation of the new decision class D_{r+1} can be computed by definition be shown as follows.

- (1) If there $[x^+] \subseteq D_{r+1}$, where $D_{r+1} = \{x^+\}$, then $\underline{R}_A^{\geq}(D_{r+1}) = [x^+]_A^{\geq}$. Otherwise $\underline{R}_A^{\geq}(D_{r+1}) = \emptyset$.
- (2) $\underline{R}_A^{\geq}(D_{r+1}) = \cup\{[x]_A^{\geq} | x^+ \in [x]_A^{\geq}, x \in U\}$.

Proof:

It's easy to prove according to the definition of approximations. □

Example 3.5. (Continued from Example 2.2). For Table 1, according to Proposition 3.5, we compute the lower and upper approximations of D_{r+1} when the object x_{15} in Table 2 be inserted into the universe U .

(1) Assume the object x_{15} be inserted into universe U in Table 2 then $U' = U \cup \{x_{15}\}$. Since for $\forall x \in U, f(x, d) \neq f(x_{15}, d) = 4$ then $U/d = \{d_1, \dots, d_r, d_{r+1}\}$ and $D_{r+1} = \{x_{15}\}$. Because of $[x_{15}]_A^{\geq} = \{x_{15}\}$, we have $\underline{R}_A^{\geq}(D_{r+1}) = \{x_{15}\}$.

(2) Let the object x_{15} in Table 2 inserts into Table 1 and $U' = U \cup \{x_{15}\}$. Since for $\forall x \in U, f(x, d) \neq f(x_{15}, d) = 4$, then $U/d = \{D_1, \dots, D_r, D_{r+1}\}$ and $D_{r+1} = \{x_{15}\}$. Because of $[x]_{15}^{\geq} \subseteq [x_2]_A^{\geq}, [x_3]_A^{\geq}, [x_4]_A^{\geq}, [x_6]_A^{\geq}, [x_9]_A^{\geq}, [x_{10}]_A^{\geq}, [x_{15}]_A^{\geq}$, we have $\underline{R}_A^{\geq}(D_{r+1}) = [x_2]_A^{\geq} \cup [x_3]_A^{\geq} \cup [x_4]_A^{\geq} \cup [x_6]_A^{\geq} \cup [x_9]_A^{\geq} \cup [x_{10}]_A^{\geq} \cup [x_{15}]_A^{\geq}, \underline{R}_A^{\geq}(D_{r+1})' = U'$.

4. Non-incremental and incremental algorithms for computing approximations in an IvOIS with the dynamic object set

In this section, we design the non-incremental and incremental algorithms on the variation of the object set in an IvOIS. Sometimes we call the non-incremental algorithm as stational algorithm or traditional algorithm.

4.1. The Non-incremental Algorithm for Computing Approximations in an IvOIS

The given Algorithm 1 is a static(non-incremental) algorithm for computing the lower and upper approximations in an IvOIS when the object set in the information system is changed. First, we compute all the decision classes $U/d = \{D_1, D_2, \dots, D_r\}$. Later, initialize all lower and upper approximations as empty set for every $D_i, i = 1, \dots, r$, The step 4-5 compute all the A -dominating sets. Step 6-15 compute the lower and upper approximations in IvOIS based on the Definition 2.2. At last, return the results. The computational complexity of Algorithm 1, as shown in Table 3.

Algorithm 1: An non-incremental algorithm for updating approximations in an IvOIS

```

Input : An decision interval-valued ordered information system  $I^\geq = (U, AT \cup \{d\}, V, f)$ .
Output : The lower and upper approximations of IvOIS
1 begin
   compute:  $U/d = \{D_1, D_2, \dots, D_r\}$ ; // the  $r$  is the cardinal number of the  $U/d$ ;
2   for  $i = 1 : r$  do
     let :  $\underline{R}_A^\geq(D_i) \leftarrow \emptyset, \overline{R}_A^\geq(D_i) \leftarrow \emptyset$  // initialize all approximations as empty set ;
3   end
4   for each  $x \in U$  do
     compute:  $[x]_A^\geq$ ; // compute all  $A$ -dominating sets with respect to all  $x$  in  $U$  classes ;
5   end
6   for  $i = 1, \dots, r$  do
7     for each  $x \in U$  do
8       if  $[x]_A^\geq \subseteq D_i$  then
9          $\underline{R}_A^\geq(D_i) = \underline{R}_A^\geq(D_i) \cup \{x\}$  // compute the lower approximation of  $D_i$  by definition ;
10        end
11        if  $x \in D_i$  then
12           $\overline{R}_A^\geq(D_i) = \overline{R}_A^\geq(D_i) \cup [x]_A^\geq$  // compute the upper approximation of  $D_i$  by definition. ;
13          end
14        end
15      end
   return :  $\underline{R}_A^\geq(D_i), \overline{R}_A^\geq(D_i)$ .
16 end

```

Table 3. The computational complexity of Algorithm 1

Step 1	$O(U ^2)$
Steps 2-3	$O(\sum_{i=1}^r D_i (U + D_i))$
Steps 4-5	$O(U ^2)$
Steps 6-15	$O(\sum_{i=1}^r U (U D_i + D_i))$
Total	$O(2 U ^2 + \sum_{i=1}^r D_i (U ^2 + U))$

4.2. The Incremental Algorithm for Updating Approximations in an IvOIS when Deleting an Object from the Universe

The given Algorithm 2 is an incremental algorithm for updating the lower and upper approximations in an IvOIS when the object set be deleted from the universe U in the interval-valued ordered informa-

tion system. Step 3-16 update the lower and upper approximations of the decision classed D_i , when the deleted object x^- belongs to the decision classes. Among them, the step 4-8 update the lower approximations of D_i by Proposition 3.1, step 9-16 update the upper approximations of D_i by Proposition 3.1. Step 18-32 update the approximations of the decision classes D_i , where the deleted object x^- does not belong to the decision classes D_i . Among them, the step 18-25 compute the lower approximations of D_i by Proposition 3.2, step 26-32 compute the upper approximations of D_i by Proposition 3.2. At last, return the result of approximations after deleting the object x^- . The computational complexity of Algorithm 2, as shown in Table 4. The flow-process diagram of Algorithm 2 as shown in Fig. 1.

Table 4. The computational complexity of Algorithm 2

Step 4-9	$O(\underline{R}_A^{\geq}(D_i))$
Steps 10-16	$O(D_i ^2 \times U)$
Steps 18-25	$O(D_i \times (U + U \times D_i))$
Steps 26-30	$O(\overline{R}_A^{\geq}(D_i))$
Total	$O(\sum_{i=1}^r D_i (\underline{R}_A^{\geq}(D_i) + 2 D_i ^2 \times U + \overline{R}_A^{\geq}(D_i)))$

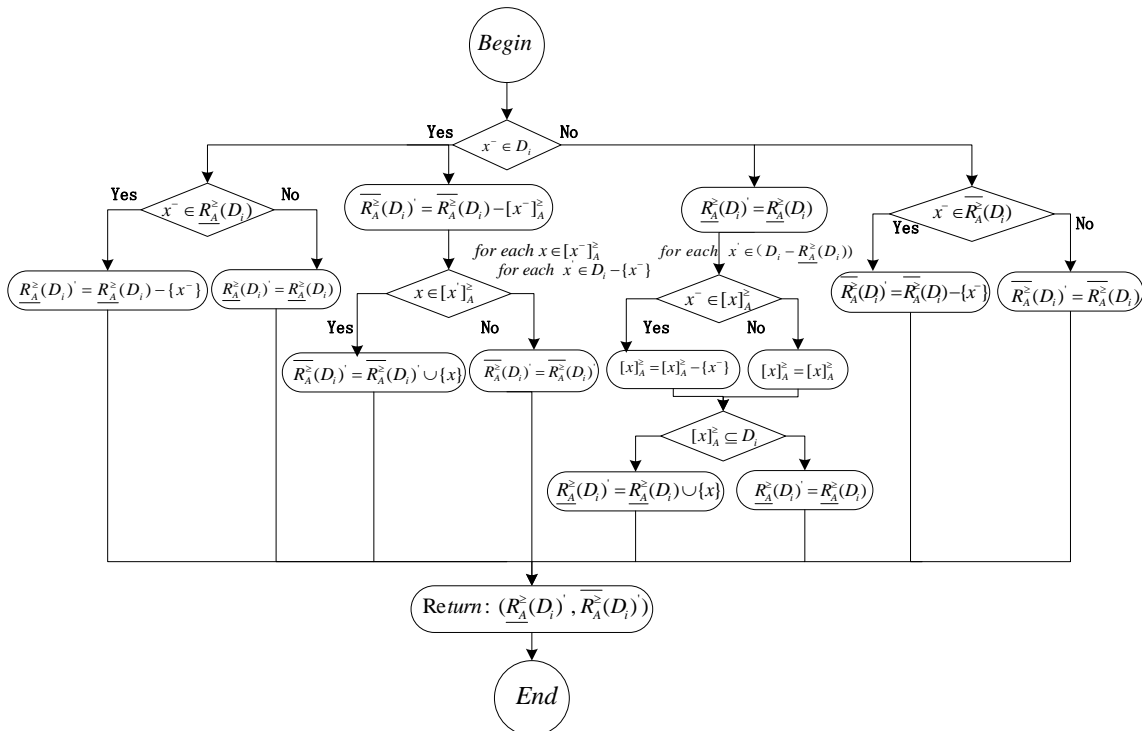


Fig. 1. The flow-process diagram of Algorithm 2.

Algorithm 2: An incremental algorithm for updating approximations in an IvOIS when deleting an object from the universe

Input :
 (1)The original interval-valued ordered information system at time $t : I^{\geq} = (U, AT \cap \{d\}, V, f)$, where $A \subseteq AT$;
 (2)The A -dominating sets $[x]_A^{\geq}$ at time t for each $x \in U$ where $A \subseteq AT$ and the original decision equivalence classes $U/d = \{D_1, D_2, \dots, D_r\}$, the r is the number of the decision classes ;
 (3)The original lower and upper approximations at time $t : \underline{R}_A^{\geq}(D_i), \overline{R}_A^{\geq}(D_i), i = 1, \dots, r$;
 (4)The object will be deleted from $U : x^-$.
Output : The lower and upper approximations in an IvOIS at time $t + 1$ after deletion of x^- from $U : \underline{R}_A^{\geq}(D_i)', \overline{R}_A^{\geq}(D_i)'$

```

1 . begin
2   for  $i = 1, \dots, r$  do
3     if  $x^- \in D_i$  then
4       if  $x^- \in \underline{R}_A^{\geq}(D_i)$  then
5         |  $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) - \{x^-\}$ ; // update the lower approximation by Proposition 3.1;
6       else
7         |  $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$ ;
8       end
9        $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i) - [x^-]_A^{\geq}$ ; //update the upper approximation by Proposition 3.1;
10      for each  $x \in [x^-]_A^{\geq}$  do
11        for each  $x' \in D_i - \{x^-\}$  do
12          if  $x \in [x']_A^{\geq}$  then
13            |  $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i)' \cup \{x\}$ ;
14          end
15        end
16      end
17    else
18      let  $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$ ; //update the lower approximation by Proposition 3.2;
19      for each  $x \in (D_i) - \overline{R}_A^{\geq}(D_i)$  do
20        if  $x^- \in [x]_A^{\geq}$  then
21          |  $[x]_A^{\geq} = [x]_A^{\geq} - \{x^-\}$ ;
22        end
23        if  $[x]_A^{\geq} \subseteq D_i$  then
24          |  $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i)' \cup \{x\}$ ;
25        end
26      end
27      if  $x^- \in \overline{R}_A^{\geq}(D_i)$  then
28        |  $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i) - \{x^-\}$ ; //update the upper approximation by Proposition 3.2;
29      else
30        |  $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i)$ ;
31      end
32    end
33  end
34  return  $\underline{R}_A^{\geq}(D_i)', \overline{R}_A^{\geq}(D_i)'$ .

```

4.3. The Incremental Algorithm for Updating Approximations in an IvOIS when Inserting an Object into the Universe

The given Algorithm 3 is an incremental algorithm for updating the lower and upper approximations in an IvOIS when the object set is inserted into the universe U in the information system. First, we should compute the A -dominating set with respect to x^+ is $[x^+]_A^{\geq}$. Step 2-22 update the approximations of decision classes D_i , when the inserted object x^+ will belong to the decision classes D_i . Step 5-9 compute the lower approximations of D_i by Proposition 3.3. Step 10 compute the upper approximations of D_i by Proposition 3.3. Step 11-22 update the approximations of the decision classes D_i , when the inserted object x^+ will not belong to the decision classes D_i . Step 11-16 compute the lower approximations of D_i by Proposition 3.4. Step 17-22 update the approximations of D_i by Proposition 3.4. Step 23-31 compute the approximation of new decision class D_{r+1} , if the inserted object does not existed decision classes. Step 23-29 compute the lower approximation of D_{r+1} by Proposition 3.5. Step 30 compute the upper approximation of D_{r+1} by Proposition 3.5. At last, return the approximations after inserting object x^+ . The computational complexity of Algorithm 3, as shown in Table 5. The flow-process diagram of Algorithm 3 as shown in Fig. 2.

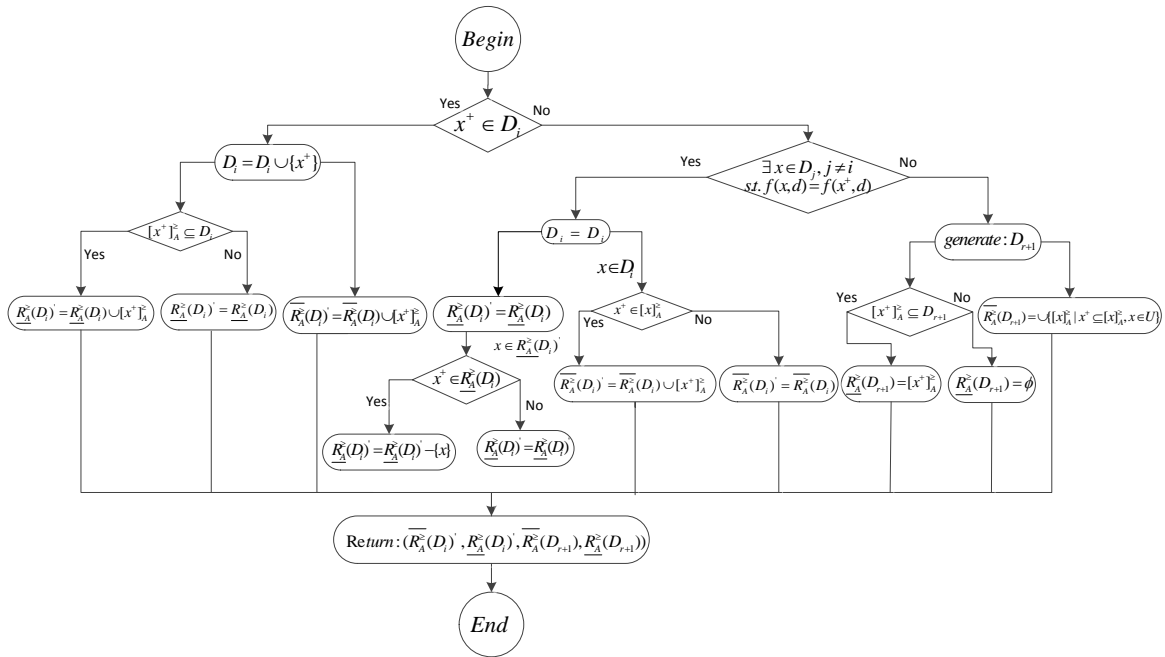


Fig. 2. The flow-process diagram of Algorithm 3.

Table 5. The computational complexity of Algorithm 3

Step 1	$O(U)$
Steps 5-10	$O(D_i \times U)$
Steps 12-16	$O(U \times R_A^{\geq}(D_i))$
Steps 17-22	$O(D_i \times U)$
Steps 23-30	$ U ^2$
Total	$O(U + \sum_{i=1}^r D_i U (2 D_i + R_A^{\geq}(D_i)) + U ^2)$

Algorithm 3: An incremental algorithm for updating approximations in an IvOIS when inserting an object into the universe U .

Input :

- (1) The original interval-valued ordered information system at time t : $I^{\geq} = (U, AT \cup \{d\}, V, f)$;
- (2) The A -dominating sets $[x]_A^{\geq}$ at time t for each $x \in U$ where $A \subseteq AT$ and the original decision equivalence classes $U/d = \{D_1, D_2, \dots, D_r\}$, the r is the number of the decision classes;
- (3) The original lower and upper approximations at time t : $\underline{R}_A^{\geq}(D_i), \overline{R}_A^{\geq}(D_i), i = 1, \dots, r$;
- (4) The object will be inserted into U : x^+ .

Output : The lower and upper approximations in an IvOIS at time $t + 1$ after the insertion of x^+ into U : $\underline{R}_A^{\geq}(D_i)', \overline{R}_A^{\geq}(D_i)'$.

```

1 begin
2   compute: the  $A$ -dominating set with respect to  $x^+$ :  $[x^+]_A^{\geq}$ ;
3   for  $i = 1, \dots, r$  do
4     if  $x^+ \in D_i$  then
5        $D_i = D_i \cup \{x^+\}$ ;
6       if  $[x^+]_A^{\geq} \subseteq D_i$  then
7          $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i) \cup \{x^+\}$ ; // update the lower approximation by Proposition 3.3;
8       else
9          $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$ ;
10      end
11       $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i) \cup [x^+]_A^{\geq}$ ; // update the upper approximation by Proposition 3.3;
12    else
13      let  $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)$ ;
14      for each  $x \in \underline{R}_A^{\geq}(D_i)'$  do
15        if  $x^+ \in [x]_A^{\geq}$  then
16           $\underline{R}_A^{\geq}(D_i)' = \underline{R}_A^{\geq}(D_i)' - \{x\}$ ; // update the lower approximation by Proposition 3.4;
17        end
18      end
19      for each  $x \in D_i$  do
20        if  $x^+ \in [x]_A^{\geq}$  then
21           $\overline{R}_A^{\geq}(D_i)' = \overline{R}_A^{\geq}(D_i)' \cup \{x^+\}$ ; // update the upper approximation by Proposition 3.4;
22        end
23      end
24    end
25    if  $\forall x \in U, f(x, d) \neq f(x^+, d)$  then
26      generate: a new decision class  $D_{r+1}$ ;
27      if  $[x^+]_A^{\geq} \subseteq D_{r+1}$  then
28         $\underline{R}_A^{\geq}(D_{r+1}) = \{x^+\}$ ; // update the lower approximation by Proposition 3.5;
29      else
30         $\underline{R}_A^{\geq}(D_{r+1}) = \emptyset$ ;
31      end
32    end
33     $\overline{R}_A^{\geq}(D_{r+1}) = \cup\{[x]_A^{\geq} | x^+ \in [x]_A^{\geq}, x \in U\}$ ; // update the upper approximation by Proposition 3.5;
34  end
35  return :  $\underline{R}_A^{\geq}(D_i)', \overline{R}_A^{\geq}(D_i)', \underline{R}_A^{\geq}(D_{r+1}), \overline{R}_A^{\geq}(D_{r+1})$ 
36 end

```

5. Case study

In this section, in order to evaluate the performance of the proposed incremental algorithms, we conduct a series of experiments to compare the computational time between the non-incremental(statical) algorithm and incremental algorithms for computing approximations based on standard data sets where from the UC Irvine Machine Learning Database Repository(<http://archive.ics.uci.edu/ml/datasets.html>), named "Energy efficiency", "Airfoil Self-Noise", "Wine Quality-red", "Wine Quality-white", "Letter Recognition", "Spoken Arabic Digit" and the characteristics of the data sets are summarized in Table 7. This experimental computing program is running on a personal computer with following hardware and software as Table 6.

Table 6. Experiment platform.

<i>Name</i>	<i>Model</i>	<i>Parameters</i>
<i>CPU</i>	<i>Intel i3 – 370</i>	<i>2.40GHz</i>
<i>Memory</i>	<i>SamsungDDR3</i>	<i>2GB, 1067MHz</i>
<i>HardDisk</i>	<i>West Data</i>	<i>500GB</i>
<i>System</i>	<i>Windows7</i>	<i>32bit</i>
<i>Platform</i>	<i>VC ++</i>	<i>6.0</i>

Note that, the attributes characteristics of the six datasets in Table 7 consist of integer or real number. We construct the interval-valued information tables by utilizing multiply error precision α , namely, the attribute value of $\forall x_i \in U$, at $\forall a_j \in AT$ is V_{x_i, a_j} , we can let it express as $[(1-\alpha) \times V_{x_i, a_j}, (1+\alpha) \times V_{x_i, a_j}]$ then we can utilize the interval-valued information tables in our experiments. In different engineering areas may use different error precision. In this paper, we set the error precision $\alpha = 0.05$.

Table 7. Experiment data sets.

No.	Data set name	Abbreviation	Objects	Attributes	Decision classes
1	<i>Energy efficiency</i>	EE	768	8	3
2	<i>Airfoil Self – Noise</i>	AS	1503	6	5
3	<i>Wine Quality – red</i>	WQ-r	1599	11	6
4	<i>Wine Quality – white</i>	WQ-w	4898	11	7
5	<i>Letter Recognition</i>	LR	8084	16	14
6	<i>Spoken ArabicDigit</i>	SAD	8800	13	3

Before the experiment let the original data in each dataset are equally divided into twenty parts. We let the original data set as the training data at time t , and randomly choose serval parts(from 5% to 50%) as the immigrating objects which will be deleted from the system at time $t + 1$. Another experiment, we choose the 80% as the training data set at time t , and the remaining 20% as the test data set which will be inserted into the system at $t + 1$. Each test choose a part enter into the system(from 10% to 100% of the test data) at time $t + 1$.

Generally, we perform the experimental analysis with applying the non-incremental algorithm along with our proposed incremental algorithms when the objects inserting into or deleting from the information system, respectively. The size of updated objects which inserting into or deleting from the universe should be different, namely, updated ratio, that is, the ration of the numbers of updating data and original data. Here, in order to analyze the influence of the updated ratio on the efficiency of algorithms, we compare the computational time of the non-incremental and incremental algorithms with different updated ratios. It's mean for each data sets, we conduct the comparison experiments with same original data size, but different updated ratios included deleting ratios and inserting ratios.

5.1. A Comparison of Computational Efficiency Between Non-incremental and Incremental Algorithm with the Deletion of the Object Set

To compare the efficiency of non-incremental algorithm (Algorithm 1) and incremental algorithm (Algorithm 2) for computing lower and upper approximations when deleting the objects from the data sets. We compute the time of the two algorithms on the given datasets in Table 7 with the different updating ratio (from 10% to 100%), but same sizes of the original data, we show the experimental results in Table 8. And more detailed changing trendline of each of two algorithms with the increasing updating ratio of data sets are presented in Fig. 3.

Table 8. A comparison of non-incremental and incremental algorithm versus different updating rates when deleting objects.

Del.(%)	EE		AS		WQ-r		WQ-w		LR		SAD	
	Non.	Incre.	Non.	Incre.	Non.	Incre.	Non.	Incre.	Non.	Incre.	Non.	Incre.
5%	0.092	0.001	0.423	0.001	1.077	0.030	10.11	0.045	60.900	0.124	25.522	0.094
10%	0.094	0.001	0.381	0.001	0.981	0.048	9.081	0.138	49.608	0.564	22.870	0.312
15%	0.062	0.001	0.375	0.048	0.888	0.096	8.097	0.234	44.116	1.372	20.436	0.656
20%	0.064	0.020	0.327	0.045	0.795	0.093	7.161	0.468	39.060	2.436	18.096	1.124
25%	0.062	0.024	0.279	0.045	0.654	0.093	6.318	0.702	34.380	3.868	15.912	1.778
30%	0.056	0.028	0.234	0.048	0.624	0.141	5.520	0.984	29.952	4.692	13.822	2.558
35%	0.052	0.026	0.195	0.048	0.516	0.141	4.728	1.404	25.832	5.488	11.950	3.494
40%	0.046	0.028	0.189	0.093	0.420	0.186	4.023	1.827	22.028	7.336	10.170	4.556
45%	0.036	0.030	0.141	0.096	0.375	0.234	3.417	2.092	18.468	8.352	8.5480	5.740
50%	0.032	0.030	0.141	0.093	0.327	0.282	2.808	2.408	15.440	11.228	7.582	6.852

In each sub-figure(a)-(f) of Fig. 3, the x -coordinate pertains to the ratio of the numbers of the deleting data and original data, while the y -coordinate concerns the computational time. According to the experimental results in Table 8 and Fig. 3, we can see, for the non-incremental algorithm, the computational time for computing approximations with deletion of the objects from the universe U is decreasing monotonically along with the increase of ratios, the size of the universe U decrease gradually. On the contrary, for the incremental algorithm, we can see that the computational efficiency for computing approximations is changing smoothly along with the increase of deleting ratios.

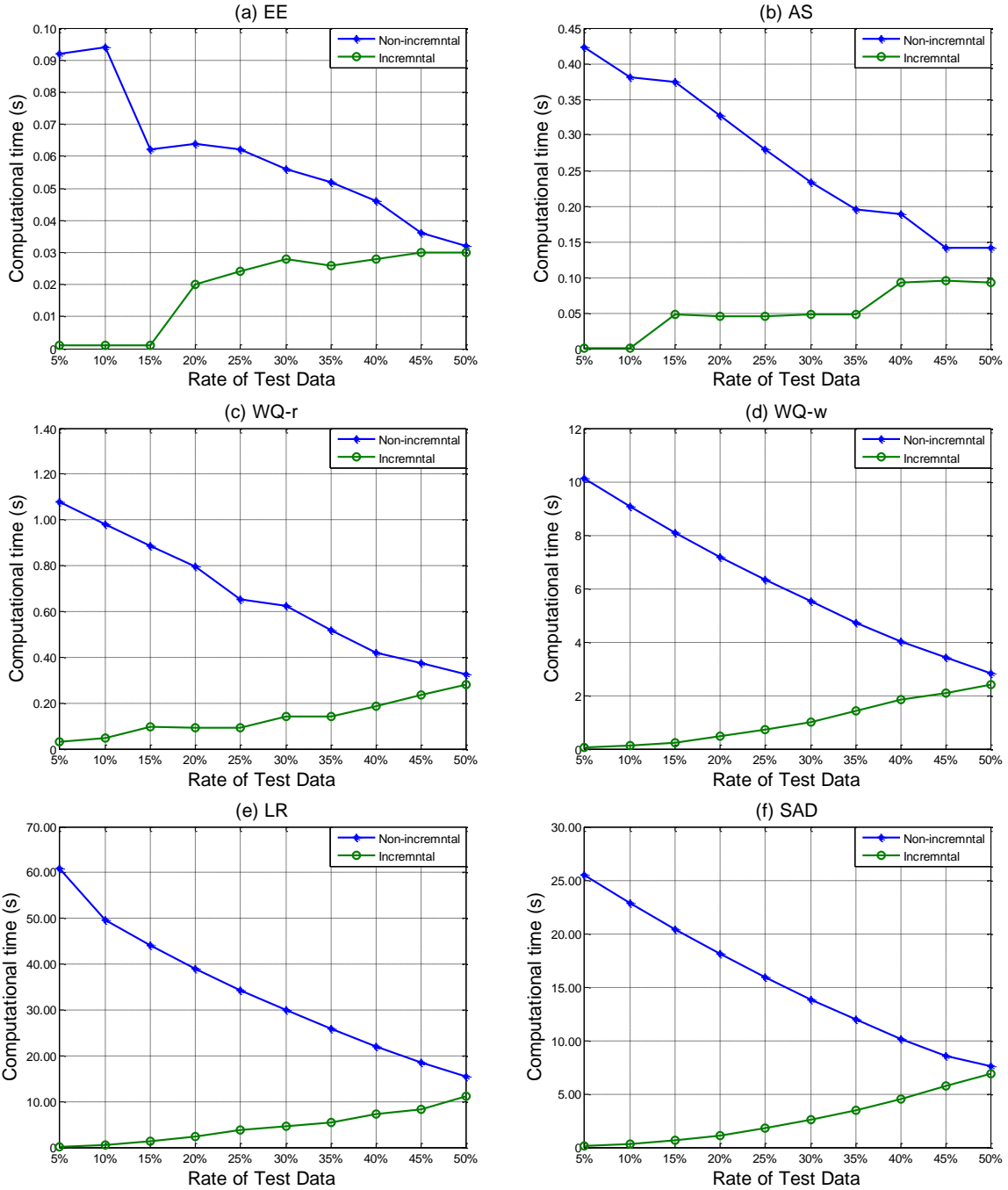


Fig. 3. A comparison of non-incremental (Algorithm 1) and incremental (Algorithm 2) algorithms versus different updating rates when deleting objects.

It's easy to get the incremental algorithm always performs faster than the non-incremental algorithm for computing approximations. It must be note that there is a threshold depending on the data set. Different data sets have different thresholds. Once the delete ratio over the threshold, namely, the deleted data set is bigger than the remaining data set maybe the incremental algorithm is slower than the non-incremental. So, the incremental algorithm is very efficiency especially when need to delete the data set is far smaller than the original data set. Data set is larger when stronger regularity.

5.2. A Comparison of Computational Efficiency Between Non-incremental and Incremental Algorithm with the Insertion of the Object Set

Similar to the experiment schemes for comparing the efficiencies between non-incremental and incremental algorithms when deleting the objects from the universe U , we also adopt such schemes to compare the performance of algorithms on the case of inserting the objects into the universe U . We compute the two algorithms (Algorithm 1 and Algorithm 3) on the six UCI data sets in Table 7 with the changing of updating ratios for each data sets. The experimental results are shown in Table 6. More detailed change trend line of each two algorithms with the increasing ratio of data sets are given in Fig. 4.

Table 9. A comparison of non-incremental and incremental algorithm versus different updating rates when inserting objects.

Ins. (%)	EE		AS		WQ-r		WQ-w		LR		SAD	
	Non.	Incre.	Non.	Incre.	Non.	Incre.	Non.	Incre.	Non.	Incre.	Non.	Incre.
10%	0.090	0.001	0.327	0.001	0.795	0.001	7.536	0.030	41.060	0.060	19.032	0.032
20%	0.094	0.001	0.327	0.001	0.843	0.001	7.956	0.048	43.120	0.124	19.936	0.062
30%	0.092	0.001	0.327	0.001	0.891	0.001	8.283	0.048	45.180	0.248	20.904	0.092
40%	0.094	0.001	0.375	0.024	0.936	0.030	8.658	0.138	47.300	0.376	21.934	0.188
50%	0.092	0.003	0.375	0.045	0.984	0.048	9.174	0.138	49.172	0.624	22.900	0.280
60%	0.092	0.004	0.420	0.048	1.029	0.045	9.501	0.234	51.728	0.876	23.930	0.438
70%	0.094	0.004	0.420	0.045	1.029	0.048	9.873	0.234	54.040	1.188	24.960	0.560
80%	0.094	0.012	0.468	0.045	1.077	0.048	10.344	0.327	56.536	1.560	25.958	0.718
90%	0.094	0.012	0.468	0.048	1.125	0.054	10.764	0.375	58.720	2.000	27.176	0.936
100%	0.094	0.012	0.468	0.048	1.218	0.048	11.187	0.468	61.088	2.432	28.298	1.124

In each sub-figure(a)-(f) of Fig. 4, the x -coordinate pertains to the ratio of the numbers of the inserted objects and test data, while the y -coordinate concerns the computational time. According to the experimental results in Table 9 and Fig. 4, we can see, for the non-incremental algorithm, the computational time for computing approximations with insertion of the objects into the universe U is increasing monotonically along with the increase of ratios. On the contrary, for the incremental algorithm, we can see that the computational efficiency for computing approximations is changing smoothly along with the increase of inserting ratios. It's easy to get the incremental algorithm always performs faster than the non-incremental algorithm for computing approximations. So, the incremental algorithm is efficiency when the objects insert into the universe, especially the original data set is an big data set and when the changing data set relatively small is very efficiency.

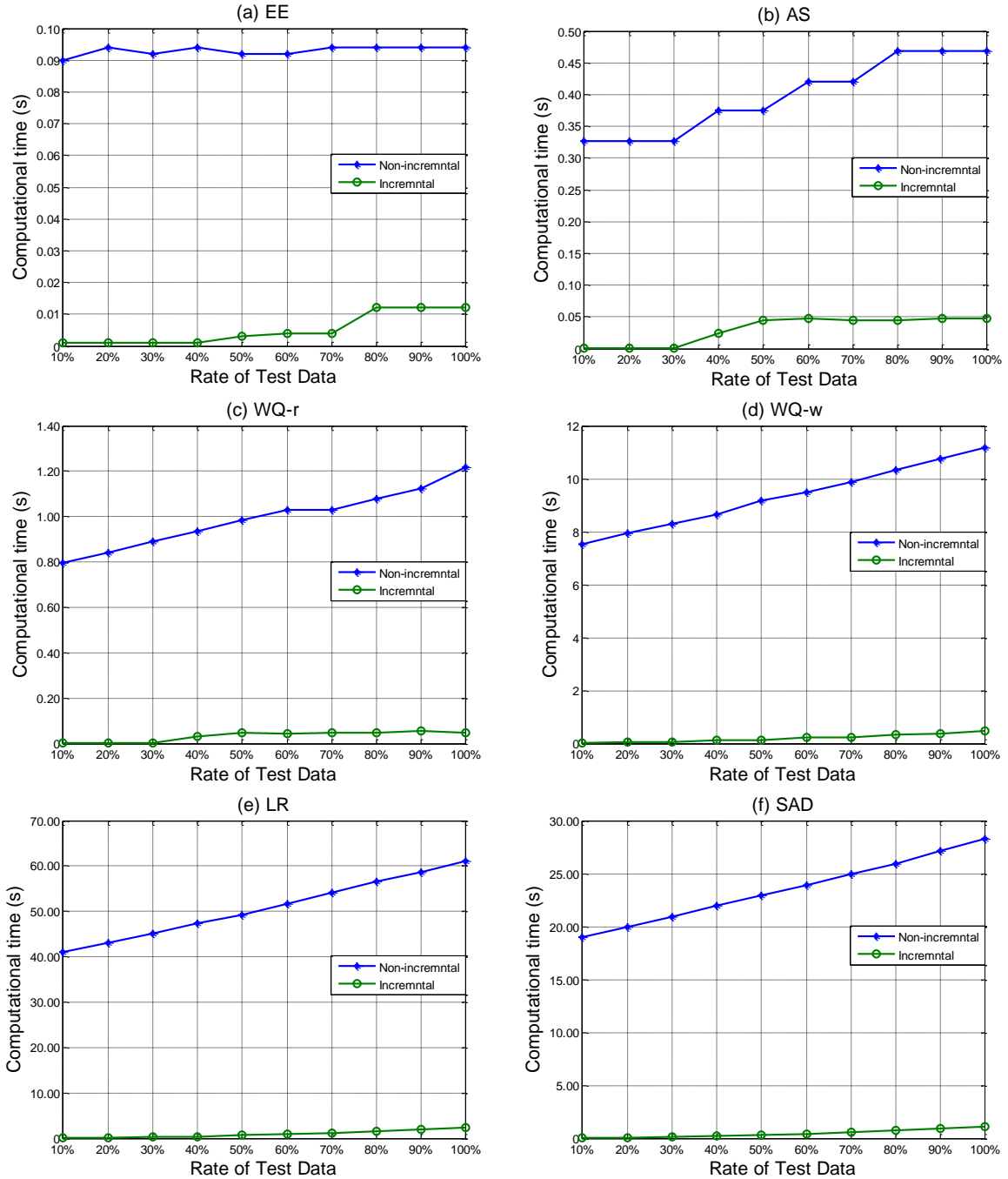


Fig. 4. A comparison of non-incremental (Algorithm 1) and incremental (Algorithm 3) algorithm versus different updating rates when inserting objects.

6. Conclusions

The incremental technique is a very effective approach to maintain knowledge in the dynamic environment. In this paper, we proposed incremental methods for updating lower and upper approximations in an IvOIS when the information system is updated by inserting or deleting object set, respectively. And two algorithms as for updating approximations when the information system is updated by inserting or deleting object set. Experimental studies pertaining to six UCI data sets showed that the incremental algorithms can improve the computational efficiency for updating approximations when the object in the information system varies over time. In real-world application, an interval-valued information system may be updated by changing granularity or attributes or all of the elements in the information system will change as time goes by under the dynamic environment. In the future, the variation of attributes and the domain of attributes values in an IvOIS or the granulations' coarsening and refinement will also be taken into consideration in terms of incremental updating approximations and knowledge discovery.

Acknowledgements

This work is supported by Natural Science Foundation of China (No. 61472463 and No. 61402064), National Natural Science Foundation of CQ CSTC (No. cstc 2013jcyjA40051), Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information (Nanjing University of Science and Technology), Ministry of Education (No. 30920140122006), Graduate Innovation Foundation of Chongqing University of Technology (No.YCX2014236), and Graduate Innovation Foundation of CQ (No.CYS15223).

References

- [1] Bang W.C., Bien Z.: New incremental learning algorithm in the framework of rough set theory, *International Journal of Fuzzy Systems*, **1**, 1999, 25-36.
- [2] N. Bryson, A. Mobolurin, An action learning evaluation procedure for multiple criteria decision making problems, *European Journal of Operational Research*, **96**, 1996, 379-386.
- [3] G. Cattaneo, D. Ciucci, Investigation about time monotonicity of similarity and preclusive rough approximations in incomplete information systems, in: *Proceedings of 4th International Conference on Rough Sets and Current Trends in Computing*, in: LNAI, **3066**, 2004, 38-48.
- [4] Y. Chen, K.W. Li, J. Levy, K.W. Hipel, D.M. Kilgour, A rough set approach to multiple criteria ABC analysis, *Transactions on Rough Sets VIII, Lecture Notes in Computer Science, Springer-Verlag, Berlin*, **5084**, 2008.
- [5] H.M. Chen, T.R. Li, S.J. Qiao, D. Ruan, A rough set based dynamic maintenance approach for approximations in coarsening and refining attribute values, *International Journal of Intelligent Systems*, **25**, 2010, 1005-1026.
- [6] H.M. Chen, T.R. Li, D. Ruan, J.H. Lin, C.X. Hu, A rough-set based incremental approach for updating approximations under dynamic maintenance environments, *IEEE Transactions on Knowledge and Data Engineering*, **25**, 2013, 274-284.
- [7] D. Ciucci, Classification of dynamics in rough sets, in: *Proceedings of 7th International Conference on Rough Sets and Current Trends in Computing*, in: LNAI, **6086**, 2010, 257-266.

- [8] D. Ciucci, Attribute dynamics in rough sets, in: *Proceedings of ISMIS11*, in: *LNCS*, **6804**, 2011, 43-51.
- [9] D. Ciucci, Temporal dynamics in information tables, *Fundamenta Informaticae*, **115**, 2012, 57-74.
- [10] Y. Fan, T. Tseng, C. Chern, C. Huang, Rule induction based on an incremental rough set, *Expert Systems with Applications* **36** 2009, 11439-11450.
- [11] S. Greco, B. Matarazzo, R. Słowiński, Handling missing values in rough set analysis of multiattribute and multi-criteria decision problems, in: *N. Zhong, A. Skowron, S. Ohsuga (Eds.), New Directions in Rough Sets, Data Mining and Granular-Soft Computing, Seventh International Workshop (RSFDGrC1999), Lecture Notes in Artificial Intelligence*, Springer-Verlag, Berlin, **1711**, 1999, 146-157.
- [12] S. Greco, B. Matarazzo, R. Słowiński, Rough approximation by dominance relations, *International Journal of Systems* **17**, 2002, 153-171.
- [13] S. Greco, B. Matarazzo, R. Słowiński, Rough sets theory for multicriteria decision analysis, *European Journal of Operational Research*, **129**, 2002, 1-47.
- [14] S. Greco, M. Inuiguchi, R. Słowiński, Fuzzy rough sets and multiple-premise gradual decision rules, *International Journal of Approximate Reasoning*, **41**, 2006, 179-211.
- [15] S. Greco, B. Matarazzo, R. Słowiński, Dominance-based rough set approach to case-based reasoning, in: *V. Torra, Y. Narukawa, A. Valls, J. Domingo-Ferrer (Eds.), Modeling Decisions for Artificial Intelligence, Third International Conference (MDAI 2006), Lecture Notes in Computer Science*, Springer-Verlag, Berlin, **3885**, 2006, 7-18.
- [16] J.W. Grzymala-Busse, Three approaches to missing attribute values: a rough set perspective, *Studies in Computational Intelligence*, **118**, 2008, 139-152.
- [17] M. Khan, M. Banerjee, Formal reasoning with rough sets in multiple-source approximation systems, *International Journal of Approximate Reasoning*, **49**, 2008, 466-477.
- [18] H.L. Li, M.H. Chen, Induction of multiple criteria optimal classification rules for biological and medical data, *Computers in Biology and Medicine*, **38**, 2008, 42-52.
- [19] J.F. Liu, Q.H. Hu, D.R. Yu, A weighted rough set based method developed for class imbalance learning, *Information Sciences*, **178**, 2008, 1235-1256.
- [20] D. Liu, T.R. Li, D. Ruan, W.L. Zou, An incremental approach for inducing knowledge from dynamic information systems, *Fundamenta Informaticae*, **94**, 2009, 245-260.
- [21] D. Liu, T.R. Li, D. Ruan, J.B. Zhang, Incremental learning optimization on knowledge discovery in dynamic business intelligent systems, *Journal of Global Optimization*, **51**, 2011, 325-344.
- [22] C. Luo, T.R. Li, H.M. Chen, D. Liu, Incremental approaches for updating approximations in set-valued ordered information systems, *Knowledge-Based Systems*, **50**, 2013, 218-233.
- [23] S.Y. Li, T.R. Li, D. Liu, Dynamic maintenance of approximations in dominance-based rough set approach under the variation of the object set, *International Journal of Systems*, **28**, 2013, 729-751.
- [24] S.Y. Li, T.R. Li, D. Liu, Incremental updating approximations in dominance-based rough sets approach under the variation of the attribute set, *Knowledge-Based systems*, **40**, 2013, 17-26.
- [25] D. Liu, D.C. Liang, Incremental learning researches on rough set theory: status and future, *International Journal of Rough Sets and Data Analysis*, **1**, 2014, 99-112.
- [26] R. Michalski, Knowledge repair mechanisms: evolution vs. revolution, in: *Proceedings of the Third International Workshop on Machine Learning*, Skytop, PA, 1985, 116-119.

- [27] E. Orlowska, Dynamic information systems, *Fundamenta Informaticae*, **5**, 1982, 101-118.
- [28] Z. Pawlak, Rough sets, *International Journal of Computer and Information Science* **11**, 1982, 341-356.
- [29] Z. Pawlak, Rough set theory and its applications to data analysis, *Cybernetics and Systems* , **29**, 1998, 661-688.
- [30] Z. Pawlak, A. Skowron, Rough sets: some extensions, *Information Sciences* , **177**, 2007, 28-40.
- [31] P. Pagliani, Pretopologies and dynamic spaces, *Fundamenta Informaticae*, **59**, 2004, 221-239.
- [32] W. Pedrycz, R. Weber, Special issue on soft computing for dynamic data mining, *Applied Soft Computing* **8**(4), 2008, 1281-1282.
- [33] Y.H. Qian, J.Y. Liang, C.Y. Dang, Interval ordered information systems, *Computers and Mathematics with Applications*, **56**, 2008, 1994-2009.
- [34] Y.H. Qian, C.Y. Dang, J.Y. Liang, D.W. Tang, Set-valued ordered information systems, *Information Sciences* , **176**(16), 2009, 2809-2832.
- [35] W.H. Xu, X.Y. Zhang, W.X. Zhang, Knowledge granulation, knowledge entropy and knowledge uncertainty measure in ordered information systems, *Applied Soft Computing*, **9**, 2009, 1244-1251.
- [36] W.H. Xu, X.Y. Zhang, J.M. Zhong, W.X. Zhang, Attribute reduction in ordered information systems based on evidence theory, *Knowledge and Information Systems*, **25**, 2010, 169-184.
- [37] W.H. Xu, Yuan Li, X.W. Liao, Approaches to attribute reductions based on rough set and matrix computation in inconsistent ordered information systems, *Knowledge-Based Systems*, **27**, 2012, 78-91.
- [38] W.H. Xu, Y.F. Liu, T.J. Li, INTUITIONISTIC FUZZY ORDERED INFORMATION SYSTEM, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **21**(3), 2013, 367-390.
- [39] X.B. Yang, D.J. Yu, J.Y. Yang, L.H. Wei, Dominance-based rough set approach to incomplete interval-valued information system, *Data & Knowledge Engineering*, **68**, 2009, 1331-1347.
- [40] X.B. Yang, Y. Qi, D.J. Yu, H.L. Yu, J.Y. Yang, α -Dominance relation and rough sets in interval-valued information systems, *Information Sciences*, **294**, 2015, 334-347.
- [41] L.Y. Zhai, L.P. Khoo, Z.W. Zhong, A dominance-based rough set approach to Kansei engineering in product development, *Expert Systems with Applications*, **36**, 2009, 393-402.
- [42] J.B. Zhang, T.R. Li, D. Ruan, D. Liu, Neighborhood rough sets for dynamic data mining, *International Journal of Intelligence Systems*, **27**, 2012, 317-342.
- [43] J.B. Zhang, J.S. Wong, T.R. Li, Y. Pan, A comparison of parallel large-scale knowledge acquisition using rough set theory on different MapReduce runtime systems, *International Journal of Approximate Reasoning*, **55**, 2014, 896-907.
- [44] Z. Zheng, G.Y. Wang, RRIA. A rough set and rule tree based incremental knowledge acquisition algorithm, *Fundamenta Informaticae* **59**, 2004, 299-313.