Knowledge-Based Systems 134 (2017) 85–104

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Incremental approaches for updating reducts in dynamic covering information systems

Guangming Lang [a,b,c], Duoqian Miao [b,c,*], Mingjie Cai [d], Zhifei Zhang [b,c]

[a] School of Mathematics and Statistics, Changsha University of Science and Technology, Changsha, Hunan 410114, PR China
[b] Department of Computer Science and Technology, Tongji University, Shanghai 201804, PR China
[c] The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, PR China
[d] College of Mathematics and Econometrics, Hunan University, Changsha, Hunan 410004, PR China

ARTICLE INFO

Article history:
Received 9 January 2017
Revised 16 June 2017
Accepted 19 July 2017
Available online 20 July 2017

Keywords:
Characteristic matrix
Covering information system
Dynamic covering information system
Rough set

ABSTRACT

In various real-world situations, there are actually a large number of dynamic covering information systems, and non-incremental learning technique is time consuming for updating approximations of sets in dynamic covering information systems. In this paper, we investigate incremental mechanisms of updating the second and sixth lower and upper approximations of sets in dynamic covering information systems with variations of attributes. Especially, we design effective algorithms for calculating the second and sixth lower and upper approximations of sets in dynamic covering information systems. The experimental results indicate that incremental algorithms outperform non-incremental algorithms in the presence of dynamic variation of attributes. Finally, we explore several examples to illustrate that the proposed approaches are feasible to perform knowledge reduction of dynamic covering information systems.

1. Introduction

Covering-based rough set theory [62] as a generalization of Pawlak's rough sets [38] is a powerful mathematical tool in the field of knowledge discovery and rule acquisition. To deal with ambiguous knowledge, researchers [28,31,35,37,61,69–71] have proposed many approximation operators and investigated their basic properties. Especially, these approximation operators are classified into three types as follows: element-based operators, granular-based operators and system-based operators, and all approximation operators are also classified into dual and non-dual operators in covering approximation spaces. Additionally, covering-based rough set theory has been applied to many fields [5,11,65] such as classification and feature selection, and related research is being increasingly studied with the development of this theory.

Many scholars [2,10,13,15,16,18,27,34,44,45,50,54–56,58–60,63,66] have investigated the lower and upper approximation operators from the view of matrix. For example, Cai et al. [2] investigated knowledge reduction of dynamic covering decision information systems caused by variations of attribute values. Hu et al. [10] proposed matrix-based dynamic approaches for updating approximations in multi-granulation rough sets when a single granular structure evolves over time. Huang et al. [13] provided an efficient approach for computing rough approximations of fuzzy concepts in dynamic fuzzy decision systems with simultaneous variation of objects and features. Lang et al. [15] presented incremental approaches for computing the second and sixth lower and upper approximations of sets from the view of matrix in dynamic covering approximation spaces. Liu [27] provided a new matrix view of rough set theory for Pawlak's lower and upper approximation operators and redefined the pair of lower and upper approximation operators using the matrix representation in a fuzzy approximation space. Ma [34] introduced fuzzy γ-covering, fuzzy γ-neighborhood, and two new types of fuzzy covering rough set models which link covering rough set theory and fuzzy rough set theory from the view of matrix. Wang et al. [50] transformed the computation of the second, fifth and sixth lower and upper approximations of a set into products of the type-1 and type-2 characteristic matrices and the characteristic function of the set in covering approximation spaces. Yang et al. [55] put forwards the fuzzy γ-minimal description and a novel type of fuzzy covering-based rough set model and investigated their properties from the view of matrix. Zhang et al. [66] presented efficient

* Corresponding author.
E-mail addresses: langguangming1984@tongji.edu.cn (G. Lang), dqmiao@tongji.edu.cn (D. Miao), cmjlong@163.com (M. Cai), zhifeizhang@tongji.edu.cn (Z. Zhang).

http://dx.doi.org/10.1016/j.knosys.2017.07.020

parallel boolean matrix based algorithms for computing rough set approximations in incomplete information systems.

Additionally, many investigations [1,3,4,6–9,12,14–16,18–26,29,30,32,33,39–43,46–49,51,53,57,58,63,64,67,68] have focused on knowledge reduction of information systems. Especially, researchers [4,15–17,20,32,36,52,63,64] have computed approximations of sets in dynamic information systems with variations of object sets, attribute sets and attribute value sets. For instance, Chen et al. [4] introduced two Boolean row vectors to characterize discernibility matrices and reducts in variable precision rough set model and employed an incremental manner to update minimal elements in the discernibility matrices at the arrival of an incremental sample. Lang et al. [16] presented incremental algorithms for computing the second and sixth lower and upper approximations of sets and investigated knowledge reduction of dynamic covering information systems with variations of objects. Li et al. [20] introduced a kind of dominance matrix to calculate P-dominating sets and P-dominated sets and discussed the principles of updating P-dominating sets and P-dominated sets when some attributes are added into or deleted from the attribute set P. Luo et al. [32] presented the updating properties for dynamic maintenance of approximations when the criteria values in set-valued decision systems evolve with time. Qian et al. [36] investigated attribute reduction for sequential three-way decisions under dynamic granulation. Zhang et al. [63] provided incremental approaches to update the relation matrix for computing the lower and upper approximations with dynamic attribute variation in set-valued information systems. Zhang et al. [64] developed incremental approaches for updating rough approximations in interval-valued information systems under attribute generalization, which carry out the computation using the previous results from the original data set along with new results.

Apart from dynamic covering information systems with variations of object sets and attribute value sets, there are many dynamic covering information systems with the immigration and emigration of attributes in practical situations, and it is time-consuming to compute the second and sixth lower and upper approximations of sets in dynamic covering information systems and perform knowledge reduction of these dynamic covering information systems with non-incremental learning technique. Furthermore, many investigations have proved that matrix-based approaches are effective to compute the lower and upper approximations of sets in dynamic information systems, and incremental approaches are more effective than non-incremental approaches for performing knowledge reduction of dynamic information systems. Therefore, we will propose incremental approaches to update the second and sixth lower and upper approximations of sets for knowledge reduction of dynamic covering information systems with variations of attribute sets from the view of matrix. The contribution of this study is shown as follows. Firstly, we study properties of the type-1 and type-2 characteristic matrices with attribute variations and provide incremental algorithms for updating the second and sixth lower and upper approximations of sets using the type-1 and type-2 characteristic matrices, respectively. Secondly, we perform experiments on ten dynamic covering information systems with attribute variations and employ the experimental results to illustrate incremental algorithms are effective to update the second and sixth lower and upper approximations of sets in dynamic covering information systems. Thirdly, we employ several examples to demonstrate that the designed algorithms are efficient to conduct knowledge reduction of dynamic covering information systems with immigrations of attributes.

The rest of this paper is organized as follows: In Section 2, we briefly review the basic concepts of Pawlak's rough set theory and covering-based rough set theory. In Section 3, we update the type-1 and type-2 characteristic matrices in dynamic covering information systems with variations of attributes. We design incremental algorithms for updating the second and sixth lower and upper approximations of sets in dynamic covering information systems. In Section 4, the experimental results illustrate incremental algorithms are effective to construct the second and sixth lower and upper approximations of sets. In Section 5, we explore several examples to illustrate how to perform knowledge reduction of dynamic covering information systems with variations of attributes. Concluding remarks and further research are given in Section 6.

2. Preliminaries

In this section, we briefly review some concepts of Pawlak's rough sets and covering-based rough sets.

Definition 2.1. [38] An information system is a 4-tuple $S=(U,A,V,f)$, where $U=\{x_1 x_2,\ldots,x_n\}$ is a finite set of objects, A is a finite set of attributes, $V=\{V_a|a\in A\}$, where V_a is the set of values of attribute a, and $card(V_a)>1$, f is a function from $U\times A$ into V.

Information systems, where objects are measured by using a finite number of attributes, represent all available knowledge and information. In practice, if we have $A=C\cup D$, where C and D denote a non-empty finite set of conditional attributes and a non-empty finite set of decision attributes, respectively, then $S=(U,A,V,f)$ is called a decision information system.

Definition 2.2. [38] Let $S=(U,A,V,f)$ be an information system, and $B\subseteq C$. Then an indiscernibility relation $IND(B)\subseteq U\times U$ is defined as:

$$IND(B)=\{(x,y)\in U\times U|\forall b\in B, b(x)=b(y)\},$$

where $b(x)$ *and* $b(y)$ denote the values of objects x and y on b, respectively.

According to Definition 2.2, we see that x and y are indiscernible with respect to B if $(x,y)\in IND(B)$ for $x,y\in U$, and obtain the family of equivalence classes $U/B=\{[x]_B|x\in U\}$ of the universe U, where $[x]_B=\{y\in U|(x,y)\in IND(B)\}$ and U/B is a partition of the universe U. Moreover, if $IND(C)\subseteq IND(D)$, then S is called a consistent information system. Otherwise, it is inconsistent.

By Definition 2.2, Pawlak presented concepts of lower and upper approximations of sets and reducts as follows.

Definition 2.3. [38] Let $S=(U,A,V,f)$ be an information system, and $B\subseteq C$. Then Pawlak's upper and lower approximations of $X\subseteq U$ with respect to $IND(B)$ are defined as follows:

$$\overline{R}(X)=\{x\in U|[x]_B\cap X\neq\emptyset\}, \underline{R}(X)=\{x\in U|[x]_B\subseteq X\}.$$

According to Definition 2.3, we see that Pawlak's lower and upper approximation operators are constructed on an indiscernibility relation or a family of equivalence classes, which are the classical approximation operators for knowledge discovery of information systems.

Definition 2.4. [38] Let $S=(U,A,V,f)$ be an information system. Then an attribute set $P\subseteq C$ is called a reduct if (1) IND(P)=IND(C); (2) For any $a\in P$, $IND(P-\{a\})\neq IND(C)$.

The first condition indicates the joint sufficiency of the attribute set P; the second condition means that each attribute in P is individually necessary. Therefore, P is the minimum attribute set keeping the indiscernibility relation $IND(C)$.

In practical situations, the condition of the indiscernibility relation is so strict that limits its applications for knowledge discovery of information systems, and Zakowski provided covering-based rough set theory for improving the performance of Pawlak's model as follows.

Definition 2.5. [62] Let U be a finite universe of discourse, and $\mathscr{C}$ a family of subsets of U. Then $\mathscr{C}$ is called a covering of U if none of elements of $\mathscr{C}$ is empty and $\bigcup\{C|C\in\mathscr{C}\}=U$. Furthermore, $(U,\mathscr{C})$ is referred to as a covering approximation space.

By Definition 2.5, we see that a covering is an extension of a partition. Furthermore, if U is a finite universe of discourse, and $\mathscr{D}=\{\mathscr{C}_1,\mathscr{C}_2,\ldots,\mathscr{C}_m\}$, where $\mathscr{C}_i(1\leq i\leq m)$ is a covering of U, then $(U,\mathscr{D})$ is called a covering information system. Especially, if all coverings of $\mathscr{D}$ are classified into conditional attribute-based coverings and decision attribute-based coverings, then $(U,\mathscr{D})$ is called a covering decision information system.

Example 2.6. Let $U=\{x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8\}$ be eight cars, $C=\{price, quality\}$ the conditional attribute set, $D=\{grade\}$ the decision attribute set, the domains of *price, quality*, and *grade* are {*high, middle, low*}, {*good, bad*}, and {1, 2}, respectively. To evaluate these cars, specialists A and B are employed and their evaluation reports are shown as follows:

$$high_A=\{x_1,x_4,x_5,x_7\}, middle_A=\{x_2,x_8\}, low_A=\{x_3,x_6\};$$

$$high_B=\{x_1,x_2,x_4,x_7,x_8\}, middle_B=\{x_5\}, low_B=\{x_3,x_6\};$$

$$good_A=\{x_1,x_2,x_3,x_6\}, bad_A=\{x_4,x_5,x_7,x_8\};$$

$$good_B=\{x_1,x_2,x_3,x_5\}, bad_B=\{x_4,x_6,x_7,x_8\},$$

where $high_*$ denotes the cars belonging to high price by the specialist $*$, and the other symbols are similar. Since their evaluations are of equal importance, we should consider all their advice. Consequently, we obtain the following results:

$$high_{A\vee B}=high_A\cup high_B=\{x_1,x_2,x_4,x_5,x_7,x_8\};$$

$$middle_{A\vee B}=middle_A\cup middle_B=\{x_2,x_5,x_8\};$$

$$low_{A\vee B}=low_A\cup low_B=\{x_3,x_6\};$$

$$good_{A\vee B}=good_A\cup good_B=\{x_1,x_2,x_3,x_5,x_6\};$$

$$bad_{A\vee B}=bad_A\cup bad_B=\{x_4,x_5,x_6,x_7,x_8\},$$

and derive a covering information system $(U,\mathscr{D}_C)$, where $\mathscr{D}_C=\{\mathscr{C}_{price},\ \mathscr{C}_{color}\}$, $\mathscr{C}_{price}=\{high_{A\vee B}, middle_{A\vee B},\ low_{A\vee B}\}$ and $\mathscr{C}_{color}=\{good_{A\vee B}, bad_{A\vee B}\}$. Furthermore, if $grade_1=\{x_1,x_2,x_3\}$ and $grade_2=\{x_4,x_5,x_6,x_7,x_8\}$, where $grade_i$ denotes the cars belonging to the *ith* grade, then we have a covering decision information system $(U,\mathscr{D}_C\cup\mathscr{D}_D)$, where $\mathscr{D}_D=\{grade_1,grade_2\}$.

Definition 2.7. [71] Let $(U,\mathscr{C})$ be a covering approximation space, and $N(x)=\bigcap\{C_i|x\in C_i\in\mathscr{C}\}$ for $x\in U$. Then the second and sixth upper and lower approximations of $X\subseteq U$ with respect to $\mathscr{C}$ are defined as follows:

(1) $SH_{\mathscr{C}}(X)=\bigcup\{C\in\mathscr{C}|C\cap X\neq\emptyset\}$, $SL_{\mathscr{C}}(X)=[SH_{\mathscr{C}}(X^c)]^c$;

(2) $XH_{\mathscr{C}}(X)=\{x\in U|N(x)\cap X\neq\emptyset\}$, $XL_{\mathscr{C}}(X)=\{x\in U|N(x)\subseteq X\}$.

According to Definition 2.7, we see that the second and sixth lower and upper approximation operators are important tools for knowledge reduction of covering information systems; and while they are typical representatives of dual and non-dual approximation operators for covering approximation spaces. They are also typical representatives of element and granular-based operators for covering approximation spaces. Moreover, since $x\in N(x)$ for any $x\in U$, we have the induced covering $\{N(x)|x\in U\}$ for the universe U, and the covering $\{N(x)|x\in U\}$ is finer than $\mathscr{C}$. Especially, the second and sixth lower and upper approximation operators are typical representatives of the original and induced coverings-based approximation operators in covering approximation spaces.

If $U=\{x_1,x_2,\ldots,x_n\}$ is a finite universe, $\mathscr{C}=\{C_1,C_2,\ldots,C_m\}$ a family of subsets of U, and $M_{\mathscr{C}}=(a_{ij})_{n\times m}$, where $a_{ij}=\begin{cases}1, & x_i\in C_j;\\ 0, & x_i\notin C_j.\end{cases}$, then $M_{\mathscr{C}}$ is called a matrix representation of $\mathscr{C}$. Additionally, we have the vector representation of characteristic function $\mathcal{X}_X=[a_1\ \ a_2\ \ .\ \ .\ \ .\ \ a_n]^T$ for $X\subseteq U$, where $a_i=\begin{cases}1, & x_i\in X;\\ 0, & x_i\notin X.\end{cases}$

Definition 2.8. [50] Let $(U,\mathscr{C})$ be a covering approximation space, $A=(a_{ij})_{n\times m}$ and $B=(b_{ij})_{m\times p}$ Boolean matrices, and $A\odot B=(c_{ij})_{n\times p}$, where $c_{ij}=\bigwedge_{k=1}^{m}(b_{kj}-a_{ik}+1)$. Then

(1) $\Gamma(\mathscr{C})=M_{\mathscr{C}}\bullet M_{\mathscr{C}}^T=(d_{ij})_{n\times n}$ is called the type-1 characteristic matrix of $\mathscr{C}$, where $d_{ij}=\bigvee_{k=1}^{m}(a_{ik}\cdot a_{jk})$, and $M_{\mathscr{C}}\bullet M_{\mathscr{C}}^T$ is the boolean product of $M_{\mathscr{C}}$ and its transpose $M_{\mathscr{C}}^T$;

(2) $\prod(\mathscr{C})=M_{\mathscr{C}}\odot M_{\mathscr{C}}^T=(e_{ij})_{n\times n}$ is called the type-2 characteristic matrix of $\mathscr{C}$.

According to Definition 2.8, we see that the type-1 and type-2 characteristic matrices provide effective tools for computing the second and sixth lower and upper approximations of sets in covering approximation spaces from the view of matrix.

We show the second and sixth lower and upper approximations of sets using the type-1 and type-2 characteristic matrices respectively as follows.

Definition 2.9. [50] Let $(U,\mathscr{C})$ be a covering approximation space, and $\mathcal{X}_X$ the characteristic function of X in U. Then

(1) $\mathcal{X}_{SH_{\mathscr{C}}(X)}=\Gamma(\mathscr{C})\bullet\mathcal{X}_X$, $\mathcal{X}_{SL_{\mathscr{C}}(X)}=\Gamma(\mathscr{C})\odot\mathcal{X}_X$; (2) $\mathcal{X}_{XH_{\mathscr{C}}(X)}=\prod(\mathscr{C})\bullet\mathcal{X}_X$, $\mathcal{X}_{XL_{\mathscr{C}}(X)}=\prod(\mathscr{C})\odot\mathcal{X}_X$.

According to Definition 2.9, we find that the computations of the second and sixth lower and upper approximations of sets in covering approximation spaces are transformed into products of the type-1 and type-2 characteristic matrices and the vector representations of the sets, respectively.

We present concepts of the type-1 and type-2 reducts of covering information systems as follows.

Definition 2.10. [15] Let $(U,\mathscr{D}_C\cup\mathscr{D}_D)$ be a covering information system, where $\mathscr{D}_C=\{\mathscr{C}_i|i\in I\}$, $\mathscr{D}_D=\{D_i|i\in J\}$, I and J are indexed sets. Then $\mathscr{P}\subseteq\mathscr{D}_C$ is called the type-1 reduct of $(U,\mathscr{D}_C\cup\mathscr{D}_D)$ if it satisfies (1) and (2) as follows:

(1) $\Gamma(\mathscr{D}_C)\bullet M_{\mathscr{D}_D}=\Gamma(\mathscr{P})\bullet M_{D_i},\Gamma(\mathscr{D}_C)\odot M_{\mathscr{D}_D}=\Gamma(\mathscr{P})\odot M_{\mathscr{D}_D}$;

(2) $\Gamma(\mathscr{D}_C)\bullet M_{\mathscr{D}_D}\neq\Gamma(\mathscr{P}')\bullet M_{\mathscr{D}_D},\Gamma(\mathscr{D}_C)\odot M_{\mathscr{D}_D}\neq\Gamma(\mathscr{P}')\odot M_{\mathscr{D}_D},\forall\mathscr{P}'\subset\mathscr{P}$.

According to Definition 2.10, we see that type-1 reducts not only keep the second lower approximations of equivalence classes but also remain the second upper approximations of equivalence classes, and it transforms the construction of reducts into products of the type-1 characteristic matrices and the vector representations of the equivalence classes. Especially, a type-1 reduct is the minimal set which remains the second lower and upper approximations of equivalence classes.

Definition 2.11. [15] Let $(U,\mathscr{D}_C\cup\mathscr{D}_D)$ be a covering information system, where $\mathscr{D}_C=\{\mathscr{C}_i|i\in I\}$, $\mathscr{D}_D=\{D_i|i\in J\}$, I and J are indexed sets. Then $\mathscr{P}\subseteq\mathscr{D}_C$ is called the type-2 reduct of $(U,\mathscr{D}_C\cup\mathscr{D}_D)$ if it satisfies (1) and (2) as follows:

(1) $\prod(\mathscr{D}_C)\bullet M_{\mathscr{D}_D}=\prod(\mathscr{P})\bullet M_{\mathscr{D}_D},\prod(\mathscr{D}_C)\odot M_{\mathscr{D}_D}=\prod(\mathscr{P})\odot M_{\mathscr{D}_D}$;

(2) $\prod(\mathscr{D}_C)\bullet M_{\mathscr{D}_D}\neq\prod(\mathscr{P}')\bullet M_{\mathscr{D}_D}, \prod(\mathscr{D}_C)\odot M_{\mathscr{D}_D}\neq\prod(\mathscr{P}')\odot M_{\mathscr{D}_D}, \forall\mathscr{P}'\subset\mathscr{P}$.

According to Definition 2.11, we observe that type-2 reducts not only keep the sixth lower approximations of equivalence classes but also remain the sixth upper approximations of equivalence classes, and it transforms the construction of reducts into products of the type-2 characteristic matrices and the vector representations of the equivalence classes. Especially, a type-2 reduct is the minimal set which keeps the sixth lower and upper approximations of equivalence classes.

Finally, we put forwards the concept of a sub-covering information system as follows.

Definition 2.12. Let $(U,\mathscr{D})$ be a covering information system, and $\mathscr{D}^j\subseteq\mathscr{D}$, where $\mathscr{D}^j$ is a family of coverings of U. Then $(U,\mathscr{D}^j)$ is called a sub-covering information system of $(U,\mathscr{D})$.

By Definition 2.12, we see that there are $2^{|\mathscr{D}|}-1$ sub-covering covering information systems for the covering information system $(U,\mathscr{D})$. Furthermore, we employ an example to illustrate the relationship between a covering information system and its sub-covering covering information systems.

Example 2.13. Let $(U,\mathscr{D})$ be a covering information system, where $U=\{x_1,x_2,x_3,x_4,x_5\}$, $\mathscr{D}=\{\mathscr{C}_1,\mathscr{C}_2,\mathscr{C}_3\}$, $\mathscr{C}_1=\{\{x_1,x_2,x_3,x_4\},\{x_5\}\}$, $\mathscr{C}_2=\{\{x_1,x_2\},\{x_3,x_4,x_5\}\}$, and $\mathscr{C}_3=\{\{x_1,x_2,x_5\},\{x_3,x_4\}\}$. Then we obtain a sub-covering information system $(U,\mathscr{D}^1)$, where $\mathscr{D}^1=\{\mathscr{C}_1,\mathscr{C}_2\}$. Furthermore, $(U,\mathscr{D}^2)$ is a sub-covering information system of $(U,\mathscr{D})$, where $\mathscr{D}^2=\{\mathscr{C}_1,\mathscr{C}_3\}$.

3. Update the type-1 and type-2 characteristic matrices in dynamic covering information systems

In this section, we provide incremental approaches to compute the type-1 and type-2 characteristic matrices in dynamic covering information systems.

Definition 3.1. Let $(U,\mathscr{D})$ and $(U,\mathscr{D}^+)$ be covering information systems, where $U=\{x_1,x_2,\ldots,x_n\}$, $\mathscr{D}=\{\mathscr{C}_1,\mathscr{C}_2,\ldots,\mathscr{C}_m\}$, and $\mathscr{D}^+=\{\mathscr{C}_1,\mathscr{C}_2,\ldots,\mathscr{C}_m,\mathscr{C}_{m+1}\}(m\geq 1)$. Then $(U,\mathscr{D}^+)$ is called a dynamic covering information system of $(U,\mathscr{D})$.

In practice, we have $|\mathscr{D}^+|=|\mathscr{D}|+1$, and $|\mathscr{D}^+|=|\mathscr{D}|+l(l\geq 2)$ when adding attributes, where $|*|$ stands for the cardinality of $*$. For the sake of simplicity, we only discuss the dynamic covering information system $(U^+,\mathscr{D}^+)$ given by Definition 3.1, and $(U,\mathscr{D})$ is called the original covering information system of $(U,\mathscr{D}^+)$.

Definition 3.2. Let $(U,\mathscr{D})$ be a covering information system, where $U=\{x_1,x_2,\ldots,x_n\}$, $\mathscr{D}=\{\mathscr{C}_1,\mathscr{C}_2,\ldots,\mathscr{C}_m\}$, and $M_{\mathscr{C}_i}$ is the matrix representation of $\mathscr{C}_i\in\mathscr{D}$. Then the matrix representation of $\mathscr{D}$ is defined as follows:

$$M_{\mathscr{D}}=[M_{\mathscr{C}_1}\quad M_{\mathscr{C}_2}\quad .\quad .\quad .\quad M_{\mathscr{C}_m}].$$

Actually, the set $\mathscr{D}$ is a family of coverings of U, and $\bigcup_{i=1}^m\{C|C\in\mathscr{C}_i\in\mathscr{D}\}$ is a covering of U, and $M_{\mathscr{D}}$ provides a representation for the family of coverings from the view of matrix, and it is useful for computing the second and sixth lower and upper approximations of sets in dynamic covering information systems.

Example 3.3. Let $(U,\mathscr{D})$ be the original covering information system, where $U=\{x_1,x_2,x_3,x_4,x_5\}$, $\mathscr{D}=\{\mathscr{C}_1,\mathscr{C}_2,\mathscr{C}_3\}$, $\mathscr{D}^+=\{\mathscr{C}_1,\mathscr{C}_2,\mathscr{C}_3,\mathscr{C}_4\}$, $\mathscr{C}_1=\{\{x_1,x_2,x_3,x_4\},\{x_5\}\}$, $\mathscr{C}_2=\{\{x_1,x_2\},\{x_3,x_4,x_5\}\}$, $\mathscr{C}_3=\{\{x_1,x_2,x_5\},\{x_3,x_4\}\}$, and $\mathscr{C}_4=\{\{x_1,x_2\},\{x_3,x_4\},\{x_5\}\}$. Then we have the matrix representations $M_{\mathscr{C}_1},M_{\mathscr{C}_2}$, $M_{\mathscr{C}_3}$, and $M_{\mathscr{C}_4}$ as follows:

$$M_{\mathscr{C}_1}=\begin{bmatrix}1&0\\1&0\\1&0\\1&0\\0&1\end{bmatrix}, M_{\mathscr{C}_2}=\begin{bmatrix}1&0\\1&0\\0&1\\0&1\\0&1\end{bmatrix},$$
$$M_{\mathscr{C}_3}=\begin{bmatrix}1&0\\1&0\\0&1\\0&1\\1&0\end{bmatrix}, M_{\mathscr{C}_4}=\begin{bmatrix}1&0&0\\1&0&0\\0&1&0\\0&1&0\\0&0&1\end{bmatrix}.$$

By Definition 3.2, we get the matrix representations $M_{\mathscr{D}}$ and $M_{\mathscr{D}^+}$ as follows:

$$M_{\mathscr{D}}=\begin{bmatrix}M_{\mathscr{C}_1}\\M_{\mathscr{C}_2}\\M_{\mathscr{C}_3}\end{bmatrix}^T=\begin{bmatrix}1&0&1&0&1&0\\1&0&1&0&1&0\\1&0&0&1&0&1\\1&0&0&1&0&1\\0&1&0&1&1&0\end{bmatrix},$$
$$M_{\mathscr{D}^+}=\begin{bmatrix}M_{\mathscr{C}_1}\\M_{\mathscr{C}_2}\\M_{\mathscr{C}_3}\\M_{\mathscr{C}_4}\end{bmatrix}^T=\begin{bmatrix}1&0&1&0&1&0&1&0&0\\1&0&1&0&1&0&1&0&0\\1&0&0&1&0&1&0&1&0\\1&0&0&1&0&1&0&1&0\\0&1&0&1&1&0&0&0&1\end{bmatrix}.$$

In what follows, we show how to construct $\Gamma(\mathscr{D}^+)$ based on $\Gamma(\mathscr{D})$. For convenience, we denote $M_{\mathscr{D}}=\begin{bmatrix}M_{\mathscr{C}_1} & M_{\mathscr{C}_2} & . & . & . & M_{\mathscr{C}_m}\end{bmatrix}$, $M_{\mathscr{D}^+}=[M_{\mathscr{C}_1}\quad M_{\mathscr{C}_2}\quad .\quad .\quad .\quad M_{\mathscr{C}_m}\quad M_{\mathscr{C}_{m+1}}]$, $M_{\mathscr{C}_k}=(a_{ij}^k)_{n\times|\mathscr{C}_k|}$, $\Gamma(\mathscr{D})=(b_{ij})_{n\times n}$, and $\Gamma(\mathscr{D}^+)=(c_{ij})_{n\times n}$.

Theorem 3.4. *Let $(U,\mathscr{D}^+)$ be a dynamic covering information system of $(U,\mathscr{D})$, $\Gamma(\mathscr{D})$ and $\Gamma(\mathscr{D}^+)$ the type-1 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^+$, respectively, and $\Gamma(\mathscr{C}_{m+1})=M_{\mathscr{C}_{m+1}}\bullet M^T_{\mathscr{C}_{m+1}}$. Then*

$$\Gamma(\mathscr{D}^+)=\Gamma(\mathscr{D})\bigvee\Gamma(\mathscr{C}_{m+1}).$$

Proof. By Definitions 2.8 and 3.1, we get $\Gamma(\mathscr{C})$ and $\Gamma(\mathscr{C}^{+})$ as follows:

$$\begin{aligned}
\Gamma(\mathscr{D}) &= M_{\mathscr{D}} \bullet M_{\mathscr{D}}^{T}\\
&= \begin{bmatrix}
a^1_{11} & a^1_{12} & \cdot & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & \cdot & a^2_{1|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{11} & a^m_{12} & \cdot & a^m_{1|\mathscr{C}_m|}\\
a^1_{21} & a^1_{22} & \cdot & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & \cdot & a^2_{2|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{21} & a^m_{22} & \cdot & a^m_{2|\mathscr{C}_m|}\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
a^1_{n1} & a^1_{n2} & \cdot & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & \cdot & a^2_{n|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{n1} & a^m_{n2} & \cdot & a^m_{n|\mathscr{C}_m|}
\end{bmatrix}\\
&\quad \bullet \begin{bmatrix}
a^1_{11} & a^1_{12} & \cdot & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & \cdot & a^2_{1|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{11} & a^m_{12} & \cdot & a^m_{1|\mathscr{C}_m|}\\
a^1_{21} & a^1_{22} & \cdot & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & \cdot & a^2_{2|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{21} & a^m_{22} & \cdot & a^m_{2|\mathscr{C}_m|}\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
a^1_{n1} & a^1_{n2} & \cdot & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & \cdot & a^2_{n|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{n1} & a^m_{n2} & \cdot & a^m_{n|\mathscr{C}_m|}
\end{bmatrix}^{T}\\
&= \begin{bmatrix}
b_{11} & b_{12} & \cdot & \cdot & \cdot & b_{1n}\\
b_{21} & b_{22} & \cdot & \cdot & \cdot & b_{2n}\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
b_{n1} & b_{n2} & \cdot & \cdot & \cdot & b_{nn}
\end{bmatrix},\\
\Gamma(\mathscr{D}^{+}) &= M_{\mathscr{D}^{+}} \bullet M_{\mathscr{D}^{+}}^{T}\\
&= \begin{bmatrix}
a^1_{11} & a^1_{12} & \cdot & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & \cdot & a^2_{1|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{11} & a^m_{12} & \cdot & a^m_{1|\mathscr{C}_m|} & a^{m+1}_{11} & a^{m+1}_{12} & \cdot & a^{m+1}_{1|\mathscr{C}_{m+1}|}\\
a^1_{21} & a^1_{22} & \cdot & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & \cdot & a^2_{2|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{21} & a^m_{22} & \cdot & a^m_{2|\mathscr{C}_m|} & a^{m+1}_{21} & a^{m+1}_{22} & \cdot & a^{m+1}_{2|\mathscr{C}_{m+1}|}\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
a^1_{n1} & a^1_{n2} & \cdot & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & \cdot & a^2_{n|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{n1} & a^m_{n2} & \cdot & a^m_{n|\mathscr{C}_m|} & a^{m+1}_{n1} & a^{m+1}_{n2} & \cdot & a^{m+1}_{n|\mathscr{C}_{m+1}|}
\end{bmatrix}\\
&\quad \bullet \begin{bmatrix}
a^1_{11} & a^1_{12} & \cdot & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & \cdot & a^2_{1|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{11} & a^m_{12} & \cdot & a^m_{1|\mathscr{C}_m|} & a^{m+1}_{11} & a^{m+1}_{12} & \cdot & a^{m+1}_{1|\mathscr{C}_{m+1}|}\\
a^1_{21} & a^1_{22} & \cdot & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & \cdot & a^2_{2|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{21} & a^m_{22} & \cdot & a^m_{2|\mathscr{C}_m|} & a^{m+1}_{21} & a^{m+1}_{22} & \cdot & a^{m+1}_{2|\mathscr{C}_{m+1}|}\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
a^1_{n1} & a^1_{n2} & \cdot & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & \cdot & a^2_{n|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{n1} & a^m_{n2} & \cdot & a^m_{n|\mathscr{C}_m|} & a^{m+1}_{n1} & a^{m+1}_{n2} & \cdot & a^{m+1}_{n|\mathscr{C}_{m+1}|}
\end{bmatrix}^{T}\\
&= \begin{bmatrix}
c_{11} & c_{12} & \cdot & \cdot & \cdot & c_{1n}\\
c_{21} & c_{22} & \cdot & \cdot & \cdot & c_{2n}\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
c_{n1} & c_{n2} & \cdot & \cdot & \cdot & c_{nn}
\end{bmatrix}.
\end{aligned}$$

According to Definition 2.8, we have

$$\begin{aligned}
b_{ij} &= \begin{bmatrix} a^1_{i1} & a^1_{i2} & \cdot & a^1_{i|\mathscr{C}_1|} & a^2_{i1} & a^2_{i2} & \cdot & a^2_{i|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{i1} & a^m_{i2} & \cdot & a^m_{i|\mathscr{C}_m|} \end{bmatrix}\\
&\quad \bullet \begin{bmatrix} a^1_{j1} & a^1_{j2} & \cdot & a^1_{j|\mathscr{C}_1|} & a^2_{j1} & a^2_{j2} & \cdot & a^2_{j|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{j1} & a^m_{j2} & \cdot & a^m_{j|\mathscr{C}_m|} \end{bmatrix}^{T}\\
&= [(a^1_{i1}\cdot a^1_{j1}) \vee (a^1_{i2}\cdot a^1_{j2}) \vee \ldots \vee (a^1_{i|\mathscr{C}_1|}\cdot a^1_{j|\mathscr{C}_1|})] \vee [(a^2_{i1}\cdot a^2_{j1}) \vee (a^2_{i2}\cdot a^2_{j2}) \vee \ldots \vee (a^2_{i|\mathscr{C}_2|}\cdot a^2_{j|\mathscr{C}_2|})] \vee \ldots \vee\\
&\quad [(a^m_{i1}\cdot a^m_{j1}) \vee (a^m_{i2}\cdot a^m_{j2}) \vee \ldots \vee (a^m_{i|\mathscr{C}_m|}\cdot a^m_{j|\mathscr{C}_m|})],\\
c_{ij} &= \begin{bmatrix} a^1_{i1} & a^1_{i2} & \cdot & a^1_{i|\mathscr{C}_1|} & a^2_{i1} & a^2_{i2} & \cdot & a^2_{i|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{i1} & a^m_{i2} & \cdot & a^m_{i|\mathscr{C}_m|} & a^{m+1}_{i1} & a^{m+1}_{i2} & \cdot & a^{m+1}_{i|\mathscr{C}_{m+1}|} \end{bmatrix}\\
&\quad \bullet \begin{bmatrix} a^1_{j1} & a^1_{j2} & \cdot & a^1_{j|\mathscr{C}_1|} & a^2_{j1} & a^2_{j2} & \cdot & a^2_{j|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{j1} & a^m_{j2} & \cdot & a^m_{j|\mathscr{C}_m|} & a^{m+1}_{j1} & a^{m+1}_{j2} & \cdot & a^{m+1}_{j|\mathscr{C}_{m+1}|} \end{bmatrix}^{T}\\
&= [(a^1_{i1}\cdot a^1_{j1}) \vee (a^1_{i2}\cdot a^1_{j2}) \vee \ldots \vee (a^1_{i|\mathscr{C}_1|}\cdot a^1_{j|\mathscr{C}_1|})] \vee [(a^2_{i1}\cdot a^2_{j1}) \vee (a^2_{i2}\cdot a^2_{j2}) \vee \ldots \vee (a^2_{i|\mathscr{C}_2|}\cdot a^2_{j|\mathscr{C}_2|})] \vee \ldots \vee\\
&\quad [(a^m_{i1}\cdot a^m_{j1}) \vee (a^m_{i2}\cdot a^m_{j2}) \vee \ldots \vee (a^m_{i|\mathscr{C}_m|}\cdot a^m_{j|\mathscr{C}_m|})] \vee [(a^{m+1}_{i1}\cdot a^{m+1}_{j1}) \vee (a^{m+1}_{i2}\cdot a^{m+1}_{j2}) \vee \ldots \vee (a^{m+1}_{i|\mathscr{C}_{m+1}|}\cdot a^{m}_{j|\mathscr{C}_{m+1}|})]\\
&= b_{ij} \vee \begin{bmatrix} a^{m+1}_{i1} & a^{m+1}_{i2} & \cdot & \cdot & \cdot & a^{m+1}_{i|\mathscr{C}_{m+1}|} \end{bmatrix} \bullet \begin{bmatrix} a^{m+1}_{j1} & a^{m+1}_{j2} & \cdot & \cdot & \cdot & a^{m+1}_{j|\mathscr{C}_{m+1}|} \end{bmatrix}^{T}.
\end{aligned}$$

To obtain $\Gamma(\mathscr{D}^+)$, we only need to compute $\Gamma(\mathscr{C}_{m+1})$ as follows:

$$\Gamma(\mathscr{C}_{m+1}) = \begin{bmatrix} a_{11}^{m+1} & a_{12}^{m+1} & . & . & . & a_{1|\mathscr{C}_{m+1}|}^{m+1} \\ a_{21}^{m+1} & a_{22}^{m+1} & . & . & . & a_{2|\mathscr{C}_{m+1}|}^{m+1} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1}^{m+1} & a_{n2}^{m+1} & . & . & . & a_{n|\mathscr{C}_{m+1}|}^{m+1} \end{bmatrix} \bullet \begin{bmatrix} a_{11}^{m+1} & a_{12}^{m+1} & . & . & . & a_{1|\mathscr{C}_{m+1}|}^{m+1} \\ a_{21}^{m+1} & a_{22}^{m+1} & . & . & . & a_{2|\mathscr{C}_{m+1}|}^{m+1} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1}^{m+1} & a_{n2}^{m+1} & . & . & . & a_{n|\mathscr{C}_{m+1}|}^{m+1} \end{bmatrix}^T.$$

Therefore, we have

$$\Gamma(\mathscr{D}^+) = \Gamma(\mathscr{D}) \bigvee \Gamma(\mathscr{C}_{m+1}),$$

where $\Gamma(\mathscr{C}_{m+1}) = M_{\mathscr{C}_{m+1}} \bullet M_{\mathscr{C}_{m+1}}^T$. □

Theorem 3.4 illustrates the mechanism of constructing $\Gamma(\mathscr{D}^+)$ based on $\Gamma(\mathscr{D})$, and it provides an effective approach to compute the second lower and upper approximations of sets in dynamic covering information systems with the immigration of attributes from the view of matrix.

We present non-incremental and incremental algorithms for computing $SH_{\mathscr{D}^+}(X)$ and $SL_{\mathscr{D}^+}(X)$ in dynamic covering information systems.

Algorithm 3.5. (Non-incremental algorithm of computing $SH_{\mathscr{D}^+}(X)$ and $SL_{\mathscr{D}^+}(X)$(**NIS**))

Step 1: Input $(U, \mathscr{D}^+)$;
Step 2: Construct $\Gamma(\mathscr{D}^+) = M_{\mathscr{D}^+} \bullet M_{\mathscr{D}^+}^T$;
Step 3: Compute $\mathcal{X}_{SH_{\mathscr{D}^+}(X)} = \Gamma(\mathscr{D}^+) \bullet \mathcal{X}_X$ and $\mathcal{X}_{SL_{\mathscr{D}^+}(X)} = \Gamma(\mathscr{D}^+) \odot \mathcal{X}_X$;
Step 4: Output $SH_{\mathscr{D}^+}(X)$ and $SL_{\mathscr{D}^+}(X)$.

Algorithm 3.6. (Incremental algorithm of computing $SH_{\mathscr{D}^+}(X)$ and $SL_{\mathscr{D}^+}(X)$(**IS**))

Step 1: Input $(U, \mathscr{D})$ and $(U, \mathscr{D}^+)$;
Step 2: Calculate $\Gamma(\mathscr{D}) = M_{\mathscr{D}} \bullet M_{\mathscr{D}}^T$;
Step 3: Construct $\Gamma(\mathscr{D}^+) = \Gamma(\mathscr{D}) \bigvee \Gamma(\mathscr{C}_{m+1})$, where $\Gamma(\mathscr{C}_{m+1}) = M_{\mathscr{C}_{m+1}} \bullet M_{\mathscr{C}_{m+1}}^T$;
Step 4: Obtain $\mathcal{X}_{SH_{\mathscr{D}^+}(X)} = \Gamma(\mathscr{D}^+) \bullet \mathcal{X}_X$ and $\mathcal{X}_{SL_{\mathscr{D}^+}(X)} = \Gamma(\mathscr{D}^+) \odot \mathcal{X}_X$;
Step 5: Output $SH_{\mathscr{D}^+}(X)$ and $SL_{\mathscr{D}^+}(X)$.

The time complexity of computing the second lower and upper approximations of sets is $O(2n^2 * \sum_{i=1}^{m+1} |\mathscr{C}_i| + 2n^2)$ using Algorithm 3.5. Furthermore, $O(2n^2 * |\mathscr{C}_{m+1}| + 3n^2)$ is the time complexity of Algorithm 3.6. Therefore, the time complexity of the incremental algorithm is lower than that of the non-incremental algorithm.

Example 3.7. (Continued from Example 3.3) Taking $X = \{x_2, x_3, x_4\}$. According to Definition 2.8, we first have

$$\Gamma(\mathscr{D}) = M_{\mathscr{D}} \bullet M_{\mathscr{D}}^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \Gamma(\mathscr{C}_4) = M_{\mathscr{C}_4} \bullet M_{\mathscr{C}_4}^T = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Second, by Theorem 3.4, we obtain

$$\Gamma(\mathscr{D}^+) = \Gamma(\mathscr{D}) \bigvee \Gamma(\mathscr{C}_4) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \bigvee \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Third, by Definition 2.9, we get

$$\mathcal{X}_{SH_{\mathscr{D}^+}(X)} = \Gamma(\mathscr{D}^+) \bullet \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \bullet \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T,$$

$$\mathcal{X}_{SL_{\mathscr{D}^+}(X)} = \Gamma(\mathscr{D}^+) \odot \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T.$$

Therefore, $SH_{\mathscr{D}^+}(X) = \{x_1, x_2, x_3, x_4, x_5\}$ and $SL_{\mathscr{D}^+}(X) = \emptyset$.

In Example 3.7, we only need to calculate elements of $\Gamma(\mathscr{C}_4)$ for computing $SH_{\mathscr{D}^+}(X)$ and $SL_{\mathscr{D}^+}(X)$ using Algorithm 3.6. But we must construct $\Gamma(\mathscr{D}^+)$ for computing $SH_{\mathscr{D}^+}(X)$ and $SL_{\mathscr{D}^+}(X)$ using Algorithm 3.5. Thereby, the incremental approach is effective to compute the second lower and upper approximations of sets.

Theorem 3.8. *Let $(U, \mathscr{D}^+)$ be a dynamic covering information system of $(U, \mathscr{D})$, $\Gamma(\mathscr{D}) = (b_{ij})_{n\times n}$ and $\Gamma(\mathscr{D}^+) = (c_{ij})_{n\times n}$ the type-1 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^+$, respectively. Then*

$$c_{ij} = \begin{cases} 1, & b_{ij} = 1; \\ \begin{bmatrix} a^{m+1}_{i1} & a^{m+1}_{i2} & . & . & . & a^{m+1}_{i|\mathscr{C}_{m+1}|} \end{bmatrix} \bullet \begin{bmatrix} a^{m+1}_{j1} & a^{m+1}_{j2} & . & . & . & a^{m+1}_{j|\mathscr{C}_{m+1}|} \end{bmatrix}^T, & b_{ij} = 0. \end{cases}$$

Proof. It is straightforward by Theorem 3.4. □

Theorem 3.8 as a simplified version of Theorem 3.4 provides an effective method for computing the type-1 characteristic matrices in dynamic covering information systems, and it is more efficient than Theorem 3.4 for computing the second lower and upper approximations of sets in dynamic covering information systems with the immigration of attributes from the view of matrix.

Example 3.9. (Continued from Example 3.7) According to Definition 2.8, we have

$$\Gamma(\mathscr{D}) = M_{\mathscr{D}} \bullet M^T_{\mathscr{D}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Therefore, by Theorem 3.8, we get

$$\Gamma(\mathscr{D}^+) = \Gamma(\mathscr{D}) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Proposition 3.10. *Let $(U, \mathscr{D}^+)$ be a dynamic covering information system of $(U, \mathscr{D})$, $\Gamma(\mathscr{D})$ and $\Gamma(\mathscr{D}^+)$ the type-1 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^+$, respectively.*

(1) *If $\Gamma(\mathscr{D}) = [1]_{n\times n}$, then $\Gamma(\mathscr{D}^+) = [1]_{n\times n}$;*
(2) *If $\Gamma(\mathscr{D}) = [0]_{n\times n}$, then $\Gamma(\mathscr{D}^+) = \Gamma(\mathscr{C}_{m+1})$.*

Proof. It is straightforward by Theorem 3.8. □

Subsequently, we construct $\prod(\mathscr{C}^+)$ based on $\prod(\mathscr{C})$. For convenience, we denote $\prod(\mathscr{C}) = (d_{ij})_{n\times n}$ and $\prod(\mathscr{C}^+) = (e_{ij})_{n\times n}$.

Theorem 3.11. *Let $(U, \mathscr{D}^+)$ be a dynamic covering information system of $(U, \mathscr{D})$, $\prod(\mathscr{D})$ and $\prod(\mathscr{D}^+)$ the type-2 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^+$, respectively, and $\prod(\mathscr{C}_{m+1}) = M_{\mathscr{C}_{m+1}} \odot M^T_{\mathscr{C}_{m+1}}$. Then*

$$\prod(\mathscr{D}^+) = \prod(\mathscr{D}) \bigwedge \prod(\mathscr{C}_{m+1}).$$

Proof. By Definitions 2.8 and 3.1, we get $\prod(\mathscr{D})$ and $\prod(\mathscr{D}^+)$ as follows:

$$\prod(\mathscr{D}) = M_{\mathscr{D}} \odot M^T_{\mathscr{D}}$$
$$= \begin{bmatrix} a^1_{11} & a^1_{12} & . & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & . & a^2_{1|\mathscr{C}_2|} & . & . & . & a^m_{11} & a^m_{12} & . & a^m_{1|\mathscr{C}_m|} \\ a^1_{21} & a^1_{22} & . & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & . & a^2_{2|\mathscr{C}_2|} & . & . & . & a^m_{21} & a^m_{22} & . & a^m_{2|\mathscr{C}_m|} \\ . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\ a^1_{n1} & a^1_{n2} & . & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & . & a^2_{n|\mathscr{C}_2|} & . & . & . & a^m_{n1} & a^m_{n2} & . & a^m_{n|\mathscr{C}_m|} \end{bmatrix}$$

$$
\odot \begin{bmatrix}
a^1_{11} & a^1_{12} & \cdot & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & \cdot & a^2_{1|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{11} & a^m_{12} & \cdot & a^m_{1|\mathscr{C}_m|} \\
a^1_{21} & a^1_{22} & \cdot & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & \cdot & a^2_{2|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{21} & a^m_{22} & \cdot & a^m_{2|\mathscr{C}_m|} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
a^1_{n1} & a^1_{n2} & \cdot & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & \cdot & a^2_{n|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{n1} & a^m_{n2} & \cdot & a^m_{n|\mathscr{C}_m|}
\end{bmatrix}^T
$$

$$
= \begin{bmatrix}
d_{11} & d_{12} & \cdot & \cdot & \cdot & d_{1n} \\
d_{21} & d_{22} & \cdot & \cdot & \cdot & d_{2n} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
d_{n1} & d_{n2} & \cdot & \cdot & \cdot & d_{nn}
\end{bmatrix},
$$

$$
\prod(\mathscr{D}^+) = M_{\mathscr{D}^+} \odot M^T_{\mathscr{D}^+}
$$

$$
= \begin{bmatrix}
a^1_{11} & a^1_{12} & \cdot & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & \cdot & a^2_{1|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{11} & a^m_{12} & \cdot & a^m_{1|\mathscr{C}_m|} & a^{m+1}_{11} & a^{m+1}_{12} & \cdot & a^{m+1}_{1|\mathscr{C}_{m+1}|} \\
a^1_{21} & a^1_{22} & \cdot & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & \cdot & a^2_{2|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{21} & a^m_{22} & \cdot & a^m_{2|\mathscr{C}_m|} & a^{m+1}_{21} & a^{m+1}_{22} & \cdot & a^{m+1}_{2|\mathscr{C}_{m+1}|} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
a^1_{n1} & a^1_{n2} & \cdot & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & \cdot & a^2_{n|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{n1} & a^m_{n2} & \cdot & a^m_{n|\mathscr{C}_m|} & a^{m+1}_{n1} & a^{m+1}_{n2} & \cdot & a^{m+1}_{n|\mathscr{C}_{m+1}|}
\end{bmatrix}
$$

$$
\odot \begin{bmatrix}
a^1_{11} & a^1_{12} & \cdot & a^1_{1|\mathscr{C}_1|} & a^2_{11} & a^2_{12} & \cdot & a^2_{1|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{11} & a^m_{12} & \cdot & a^m_{1|\mathscr{C}_m|} & a^{m+1}_{11} & a^{m+1}_{12} & \cdot & a^{m+1}_{1|\mathscr{C}_{m+1}|} \\
a^1_{21} & a^1_{22} & \cdot & a^1_{2|\mathscr{C}_1|} & a^2_{21} & a^2_{22} & \cdot & a^2_{2|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{21} & a^m_{22} & \cdot & a^m_{2|\mathscr{C}_m|} & a^{m+1}_{21} & a^{m+1}_{22} & \cdot & a^{m+1}_{2|\mathscr{C}_{m+1}|} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
a^1_{n1} & a^1_{n2} & \cdot & a^1_{n|\mathscr{C}_1|} & a^2_{n1} & a^2_{n2} & \cdot & a^2_{n|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{n1} & a^m_{n2} & \cdot & a^m_{n|\mathscr{C}_m|} & a^{m+1}_{n1} & a^{m+1}_{n2} & \cdot & a^{m+1}_{n|\mathscr{C}_{m+1}|}
\end{bmatrix}^T
$$

$$
= \begin{bmatrix}
e_{11} & e_{12} & \cdot & \cdot & \cdot & e_{1n} \\
e_{21} & e_{22} & \cdot & \cdot & \cdot & e_{2n} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
e_{n1} & e_{n2} & \cdot & \cdot & \cdot & e_{nn}
\end{bmatrix}.
$$

According to Definition 2.8, we have

$$
\begin{aligned}
d_{ij} = {} & \begin{bmatrix} a^1_{i1} & a^1_{i2} & \cdot & a^1_{i|\mathscr{C}_1|} & a^2_{i1} & a^2_{i2} & \cdot & a^2_{i|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{i1} & a^m_{i2} & \cdot & a^m_{i|\mathscr{C}_m|} \end{bmatrix} \\
& \odot \begin{bmatrix} a^1_{j1} & a^1_{j2} & \cdot & a^1_{j|\mathscr{C}_1|} & a^2_{j1} & a^2_{j2} & \cdot & a^2_{j|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{j1} & a^m_{j2} & \cdot & a^m_{j|\mathscr{C}_m|} \end{bmatrix}^T \\
= {} & [(a^1_{j1} - a^1_{i1} + 1) \wedge (a^1_{j2} - a^1_{i2} + 1) \wedge \ldots \wedge (a^1_{j|\mathscr{C}_1|} - a^1_{i|\mathscr{C}_1|} + 1)] \\
& \wedge [(a^2_{j1} - a^2_{i1} + 1) \wedge (a^2_{j2} - a^2_{i2} + 1) \wedge \ldots \wedge (a^2_{j|\mathscr{C}_2|} - a^2_{i|\mathscr{C}_2|} + 1)] \\
& \wedge \ldots \wedge [(a^m_{j1} - a^m_{i1} + 1) \wedge (a^m_{j2} - a^m_{i2} + 1) \wedge \ldots \wedge (a^m_{j|\mathscr{C}_m|} - a^m_{i|\mathscr{C}_m|} + 1)], \\
e_{ij} = {} & \begin{bmatrix} a^1_{i1} & a^1_{i2} & \cdot & a^1_{i|\mathscr{C}_1|} & a^2_{i1} & a^2_{i2} & \cdot & a^2_{i|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{i1} & a^m_{i2} & \cdot & a^m_{i|\mathscr{C}_m|} & a^{m+1}_{i1} & a^{m+1}_{i2} & \cdot & a^{m+1}_{i|\mathscr{C}_{m+1}|} \end{bmatrix} \\
& \odot \begin{bmatrix} a^1_{j1} & a^1_{j2} & \cdot & a^1_{j|\mathscr{C}_1|} & a^2_{j1} & a^2_{j2} & \cdot & a^2_{j|\mathscr{C}_2|} & \cdot & \cdot & \cdot & a^m_{j1} & a^m_{j2} & \cdot & a^m_{j|\mathscr{C}_m|} & a^{m+1}_{j1} & a^{m+1}_{j2} & \cdot & a^{m+1}_{j|\mathscr{C}_{m+1}|} \end{bmatrix}^T \\
= {} & [(a^1_{j1} - a^1_{i1} + 1) \wedge (a^1_{j2} - a^1_{i2} + 1) \wedge \ldots \wedge (a^1_{j|\mathscr{C}_1|} - a^1_{i|\mathscr{C}_1|} + 1)] \\
& \wedge [(a^2_{j1} - a^2_{i1} + 1) \wedge (a^2_{j2} - a^2_{i2} + 1) \wedge \ldots \wedge (a^2_{j|\mathscr{C}_2|} - a^2_{i|\mathscr{C}_2|} + 1)] \\
& \wedge \ldots \wedge [(a^m_{j1} - a^m_{i1} + 1) \wedge (a^m_{j2} - a^m_{i2} + 1) \wedge \ldots \wedge (a^m_{j|\mathscr{C}_m|} - a^m_{i|\mathscr{C}_m|} + 1)] \\
& \wedge [(a^{m+1}_{j1} - a^{m+1}_{i1} + 1) \wedge (a^{m+1}_{j2} - a^{m+1}_{i2} + 1) \wedge \ldots \wedge (a^{m+1}_{j|\mathscr{C}_{m+1}|} - a^{m+1}_{i|\mathscr{C}_{m+1}|} + 1)] \\
= {} & d_{ij} \wedge \begin{bmatrix} a^{m+1}_{i1} & a^{m+1}_{i2} & \cdot & \cdot & \cdot & a^{m+1}_{i|\mathscr{C}_{m+1}|} \end{bmatrix} \odot \begin{bmatrix} a^{m+1}_{j1} & a^{m+1}_{j2} & \cdot & \cdot & \cdot & a^{m+1}_{j|\mathscr{C}_{m+1}|} \end{bmatrix}^T.
\end{aligned}
$$

To obtain $\prod(\mathscr{D}^+)$, we only need to compute $\prod(\mathscr{C}_{m+1})$ as follows:

$$\prod(\mathscr{C}_{m+1}) = \begin{bmatrix} a_{11}^{m+1} & a_{12}^{m+1} & . & . & . & a_{1|\mathscr{C}_{m+1}|}^{m+1} \\ a_{21}^{m+1} & a_{22}^{m+1} & . & . & . & a_{2|\mathscr{C}_{m+1}|}^{m+1} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1}^{m+1} & a_{n2}^{m+1} & . & . & . & a_{n|\mathscr{C}_{m+1}|}^{m+1} \end{bmatrix} \odot \begin{bmatrix} a_{11}^{m+1} & a_{12}^{m+1} & . & . & . & a_{1|\mathscr{C}_{m+1}|}^{m+1} \\ a_{21}^{m+1} & a_{22}^{m+1} & . & . & . & a_{2|\mathscr{C}_{m+1}|}^{m+1} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1}^{m+1} & a_{n2}^{m+1} & . & . & . & a_{n|\mathscr{C}_{m+1}|}^{m+1} \end{bmatrix}^T.$$

Therefore, we have

$$\prod(\mathscr{D}^+) = \prod(\mathscr{D}) \bigwedge \prod(\mathscr{C}_{m+1}),$$

where $\prod(\mathscr{C}_{m+1}) = M_{\mathscr{C}_{m+1}} \odot M_{\mathscr{C}_{m+1}}^T$. □

Theorem 3.11 illustrates the mechanism of constructing $\prod(\mathscr{D}^+)$ based on $\prod(\mathscr{D})$, and it provides an effective approach to compute the type-2 characteristic matrices for knowledge reduction of dynamic covering information systems with the immigration of attributes from the view of matrix.

We provide non-incremental and incremental algorithms for computing $XH_{\mathscr{D}^+}(X)$ and $XL_{\mathscr{D}^+}(X)$ in dynamic covering information systems.

Algorithm 3.12. (Non-incremental algorithm of computing $XH_{\mathscr{D}^+}(X)$ and $XL_{\mathscr{D}^+}(X)$(**NIX**))

Step 1: Input $(U, \mathscr{D}^+)$;

Step 2: Construct $\prod(\mathscr{D}^+) = M_{\mathscr{D}^+} \odot M_{\mathscr{D}^+}^T$;

Step 3: Compute $\mathcal{X}_{XH_{\mathscr{D}^+}(X)} = \prod(\mathscr{D}^+) \bullet \mathcal{X}_X$ and $\mathcal{X}_{XL_{\mathscr{D}^+}(X)} = \prod(\mathscr{D}^+) \odot \mathcal{X}_X$;

Step 4: Output $XH_{\mathscr{D}^+}(X)$ and $XL_{\mathscr{D}^+}(X)$.

Algorithm 3.13. (Incremental algorithm of computing $XH_{\mathscr{D}^+}(X)$ and $XL_{\mathscr{D}^+}(X)$(**IX**))

Step 1: Input $(U, \mathscr{D})$ and $(U, \mathscr{D}^+)$;

Step 2: Construct $\prod(\mathscr{D}) = M_{\mathscr{D}} \odot M_{\mathscr{D}}^T$;

Step 3: Calculate $\prod(\mathscr{D}^+) = \prod(\mathscr{D}) \bigwedge \prod(\mathscr{C}_{m+1})$, where $\prod(\mathscr{C}_{m+1}) = M_{\mathscr{C}_{m+1}} \odot M_{\mathscr{C}_{m+1}}^T$;

Step 4: Get $XH_{\mathscr{D}^+}(X) = \prod(\mathscr{D}^+) \cdot \mathcal{X}_X$ and $XL_{\mathscr{D}^+}(X) = \prod(\mathscr{D}^+) \odot \mathcal{X}_X$;

Step 5: Output $XH_{\mathscr{D}^+}(X)$ and $XL_{\mathscr{D}^+}(X)$.

The time complexity of computing the sixth lower and upper approximations of sets is $O(2n^2 * \sum_{i=1}^{m+1} |\mathscr{C}_i| + 2n^2)$ by Algorithm 3.12. Furthermore, $O(2n^2 * |\mathscr{C}_{m+1}| + 3n^2)$ is the time complexity of Algorithm 3.13. Therefore, the time complexity of the incremental algorithm is lower than that of the non-incremental algorithm.

$$e_{ij} = \begin{cases} 0, & d_{ij} = 0; \\ \begin{bmatrix} a_{i1}^{m+1} & a_{i2}^{m+1} & . & . & . & a_{i|\mathscr{C}_{m+1}|}^{m+1} \end{bmatrix} \odot \begin{bmatrix} a_{j1}^{m+1} & a_{j2}^{m+1} & . & . & . & a_{j|\mathscr{C}_{m+1}|}^{m+1} \end{bmatrix}^T, & d_{ij} = 1. \end{cases}$$

Example 3.14. (Continued from Example 3.3) Taking $X = \{x_2, x_3, x_4\}$. By Definition 2.8, we first have

$$\prod(\mathscr{D}) = M_{\mathscr{D}} \odot M_{\mathscr{D}}^T = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Second, by Theorem 3.11, we get

$$\prod(\mathscr{D}^+) = \prod(\mathscr{D}) \bigwedge \prod(\mathscr{C}_4) = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \bigwedge \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Third, according to Definition 2.9, we obtain

$$\mathcal{X}_{XH_{\mathscr{D}^+}(X)} = \prod(\mathscr{D}^+) \bullet \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \end{bmatrix}^T,$$

$$\mathcal{X}_{XL_{\mathscr{D}^+}(X)} = \prod(\mathscr{D}^+) \odot \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \end{bmatrix}^T.$$

Therefore, $XH_{\mathscr{D}^+}(X) = \{x_1, x_2, x_3, x_4\}$ and $XL_{\mathscr{D}^+}(X) = \{x_3, x_4\}$.

In Example 3.14, we must compute $\prod(\mathscr{D}^+)$ for constructing $XH_{\mathscr{D}^+}(X)$ and $XL_{\mathscr{D}^+}(X)$ using Algorithm 3.12. But we only need to calculate $\prod(\mathscr{C}_4)$ for computing $XH_{\mathscr{D}^+}(X)$ and $XL_{\mathscr{D}^+}(X)$ using Algorithm 3.13. Thereby, the incremental approach is effective to compute the sixth lower and upper approximations of sets.

Theorem 3.15. *Let $(U, \mathscr{D}^+)$ be a dynamic covering information system of $(U, \mathscr{D})$, $\prod(\mathscr{C}) = (d_{ij})_{n\times n}$ and $\prod(\mathscr{C}^+) = (e_{ij})_{n\times n}$ the type-2 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^+$, respectively. Then*

Proof. It is straightforward by Theorem 3.11. □

Theorem 3.15 as a simplified version of Theorem 3.11 provides an effective method for computing the type-2 characteristic matrices in dynamic covering information systems, and it is more efficient than Theorem 3.11 for computing the sixth lower and upper approximations of sets in dynamic covering information systems with the immigration of attributes from the view of matrix.

Example 3.16. (Continued from Example 3.3) According to Definition 2.8, we have

$$\prod(\mathscr{D}) = M_{\mathscr{D}} \odot M_{\mathscr{D}}^T = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Therefore, by Theorem 3.15, we obtain

$$\prod(\mathscr{D}^+) = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Proposition 3.17. *Let $(U, \mathscr{D}^+)$ be a dynamic covering information system of $(U, \mathscr{D})$, $\prod(\mathscr{D})$ and $\prod(\mathscr{D}^+)$ the type-2 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^+$, respectively.*

(1) *If $\prod(\mathscr{D}) = [0]_{n\times n}$, then $\prod(\mathscr{D}^+) = [0]_{n\times n}$;*
(2) *If $\prod(\mathscr{D}) = [1]_{n\times n}$, then $\prod(\mathscr{D}^+) = \prod(\mathscr{C}_{m+1})$.*

Proof. It is straightforward by Theorem 3.15. □

In practical situations, there are some dynamic covering information systems with the emigration of attributes as follows.

Definition 3.18. Let $(U, \mathscr{D})$ and $(U, \mathscr{D}^-)$ be covering information systems, where $U = \{x_1, x_2, \ldots, x_n\}$, $\mathscr{D} = \{\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_m\}$, and $\mathscr{D}^- = \{\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_{m-1}\} (m \geq 2)$. Then $(U, \mathscr{D}^-)$ is called a dynamic covering information system of $(U, \mathscr{D})$.

In what follows, $(U, \mathscr{D})$ is referred to as the original covering information system of $(U, \mathscr{D}^-)$, and we employ an example to illustrate the dynamic covering information system given by Definition 3.18 as follows.

Example 3.19. Let $(U, \mathscr{D})$ be the original covering information system, where $U = \{x_1, x_2, x_3, x_4, x_5\}$, $\mathscr{D} = \{\mathscr{C}_1, \mathscr{C}_2, \mathscr{C}_3, \mathscr{C}_4\}$, $\mathscr{C}_1 = \{\{x_1, x_2, x_3, x_4\}, \{x_5\}\}$, $\mathscr{C}_2 = \{\{x_1, x_2\}, \{x_3, x_4, x_5\}\}$, $\mathscr{C}_3 = \{\{x_1, x_2, x_5\}, \{x_3, x_4\}\}$, and $\mathscr{C}_4 = \{\{x_1, x_2\}, \{x_3, x_4\}, \{x_5\}\}$. If we delete $\mathscr{C}_4$ from $\mathscr{D}$, then we obtain a dynamic covering information system $(U, \mathscr{D}^-)$ of $(U, \mathscr{D})$, where $\mathscr{D}^- = \{\mathscr{C}_1, \mathscr{C}_2, \mathscr{C}_3\}$.

We show how to construct $\Gamma(\mathscr{C}^-)$ based on $\Gamma(\mathscr{C})$. For convenience, we denote $\Gamma(\mathscr{D}) = (b_{ij})_{n\times n}$ and $\Gamma(\mathscr{D}^-) = (c^-_{ij})_{n\times n}$.

Theorem 3.20. *Let $(U, \mathscr{D}^-)$ be a dynamic covering information system of $(U, \mathscr{D})$, $\Gamma(\mathscr{D}) = (b_{ij})_{n\times n}$ and $\Gamma(\mathscr{D}^-) = (c^-_{ij})_{n\times n}$ the type-1 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^+$, respectively. Then*

$$c^-_{ij} = \begin{cases} 0, & b_{ij} = 0; \\ 1, & b_{ij} = 1 \wedge \triangle c_{ij} = 0; \\ c^*_{ij}, & b_{ij} = 1 \wedge \triangle c_{ij} = 1. \end{cases}$$

where

$$\triangle c_{ij} = \begin{bmatrix} a^m_{i1} & a^m_{i2} & . & . & . & a^m_{i|\mathscr{C}_m|} \end{bmatrix} \bullet \begin{bmatrix} a^m_{j1} & a^m_{j2} & . & . & . & a^m_{j|\mathscr{C}_m|} \end{bmatrix}^T;$$

$$c^*_{ij} = \begin{bmatrix} a^1_{i1} & a^1_{i2} & . & a^1_{i|\mathscr{C}_1|} & a^2_{i1} & a^2_{i2} & . & a^2_{i|\mathscr{C}_2|} & . & . & . & a^{m-1}_{i1} & a^{m-1}_{i2} & . & a^{m-1}_{i|\mathscr{C}_{m-1}|} \end{bmatrix}$$
$$\bullet \begin{bmatrix} a^1_{j1} & a^1_{j2} & . & a^1_{j|\mathscr{C}_1|} & a^2_{j1} & a^2_{j2} & . & a^2_{j|\mathscr{C}_2|} & . & . & . & a^{m-1}_{j1} & a^{m-1}_{j2} & . & a^{m-1}_{j|\mathscr{C}_{m-1}|} \end{bmatrix}^T.$$

Proof. It is straightforward by Theorem 3.4. □

Theorem 3.20 illustrates the mechanism of constructing $\Gamma(\mathscr{D}^-)$ based on $\Gamma(\mathscr{D})$, and it provides an effective approach to compute the second lower and upper approximations of sets in dynamic covering information systems with the emigration of attributes from the view of matrix.

Example 3.21. (Continued from Example 3.16) Taking $X = \{x_2, x_3, x_4\}$. According to Definition 2.8, we first obtain

$$\Gamma(\mathscr{D}) = M_{\mathscr{D}} \bullet M_{\mathscr{D}}^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Second, by Theorem 3.20, we get

$$\Gamma(\mathscr{D}^-) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Third, by Definition 2.9, we have

$$\mathcal{X}_{SH_{\mathscr{D}^-}(X)} = \Gamma(\mathscr{D}^-) \bullet \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \bullet \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T,$$
$$\mathcal{X}_{SL_{\mathscr{D}^-}(X)} = \Gamma(\mathscr{D}^-) \odot \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T.$$

Therefore, $SH_{\mathscr{D}^-}(X) = \{x_1, x_2, x_3, x_4, x_5\}$ and $SL_{\mathscr{D}^-}(X) = \emptyset$.

We describe how to construct $\prod(\mathscr{C}^-)$ based on $\prod(\mathscr{C})$. For convenience, we denote $\prod(\mathscr{D}) = (d_{ij})_{n\times n}$ and $\prod(\mathscr{D}^-) = (e^-_{ij})_{n\times n}$.

Theorem 3.22. *Let $(U, \mathscr{D}^-)$ be a dynamic covering information system of $(U, \mathscr{D})$, $\prod(\mathscr{D}) = (d_{ij})_{n\times n}$ and $\prod(\mathscr{D}^-) = (e^-_{ij})_{n\times n}$ the type-2 characteristic matrices of $\mathscr{D}$ and $\mathscr{D}^-$, respectively. Then*

$$e^-_{ij} = \begin{cases} 1, & d_{ij} = 1 \wedge \triangle e_{ij} = 1; \\ 0, & d_{ij} = 0 \wedge \triangle e_{ij} = 1; \\ e^*_{ij}, & d_{ij} = 0 \wedge \triangle e_{ij} = 0. \end{cases}$$

where

$$\triangle e_{ij} = \begin{bmatrix} a^m_{i1} & a^m_{i2} & . & . & . & a^m_{i|\mathscr{C}_m|} \end{bmatrix} \odot \begin{bmatrix} a^m_{j1} & a^m_{j2} & . & . & . & a^m_{j|\mathscr{C}_m|} \end{bmatrix}^T,$$

$$e^*_{ij} = \begin{bmatrix} a^1_{i1} & a^1_{i2} & . & a^1_{i|\mathscr{C}_1|} & a^2_{i1} & a^2_{i2} & . & a^2_{i|\mathscr{C}_2|} & . & . & . & a^{m-1}_{i1} & a^{m-1}_{i2} & . & a^{m-1}_{i|\mathscr{C}_{m-1}|} \end{bmatrix}$$
$$\odot \begin{bmatrix} a^1_{j1} & a^1_{j2} & . & a^1_{j|\mathscr{C}_1|} & a^2_{j1} & a^2_{j2} & . & a^2_{j|\mathscr{C}_2|} & . & . & . & a^{m-1}_{j1} & a^{m-1}_{j2} & . & a^{m-1}_{j|\mathscr{C}_{m-1}|} \end{bmatrix}^T.$$

Proof. It is straightforward by Theorem 3.11. □

Theorem 3.22 illustrates the mechanism of constructing $\prod(\mathscr{D}^-)$ based on $\prod(\mathscr{D})$, and it provides an effective approach to compute the sixth lower and upper approximations of sets in dynamic covering information systems with the emigration of attributes from the view of matrix.

Example 3.23. (Continued from Example 3.19) According to Definition 2.8, we first have

$$\prod(\mathscr{D}) = M_{\mathscr{D}} \odot M^T_{\mathscr{D}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Second, by Theorem 3.22, we get

$$\prod(\mathscr{D}^-) = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Third, by Definition 2.9, we obtain

$$\mathcal{X}_{XH_{\mathscr{D}^-}(X)} = \prod(\mathscr{D}^-) \bullet \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \end{bmatrix}^T,$$
$$\mathcal{X}_{XL_{\mathscr{D}^-}(X)} = \prod(\mathscr{D}^-) \odot \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \end{bmatrix}^T.$$

Therefore, $XH_{\mathscr{D}^-}(X) = \{x_1, x_2, x_3, x_4\}$ and $XL_{\mathscr{D}^-}(X) = \{x_3, x_4\}$.

In practical situations, we compute the type-1 and type-2 characteristic matrices for dynamic covering information systems with the immigrations and emigrations of attributes simultaneously using two steps as follows: (1) compute the type-1 and type-2 characteristic matrices by Theorems 3.4 and 3.11, respectively; (2) construct the type-1 and type-2 characteristic matrices by Theorems 3.20 and 3.22, respectively. Actually, there are more dynamic covering information systems given by Definition 3.1 than Definition 3.18. Therefore, the discussion in Section 4 focuses on dynamic covering information systems given by Definition 3.1.

4. Experimental analysis

In this section, we perform experiments to illustrate the effectiveness of Algorithms 3.6 and 3.13 for computing the second and sixth lower and upper approximations of sets, respectively, in dynamic covering information systems.

To test Algorithms 3.6 and 3.13, we generated randomly ten artificial covering information systems $\{(U_i, \mathscr{D}_i) | i = 1, 2, 3, \ldots, 10\}$, which are outlined in Table 1, where $|U_i|$ means the number of objects in U_i and $\mathscr{D}_i = \{\mathscr{C}^i_j | 1 \leq j \leq 1000\}$. For convenience, each covering here contains five elements in each covering information system $(U_i, \mathscr{D}_i)$. Moreover, we conducted all computations on a PC with a Intel(R) Dual-Core(TM) i5-4590 CPU @ 3.30 GHZ and 8 GB memory, running 64-bit Windows 7; the software was 64-bit Matlab R2009b.

Remark. There are two reasons to choose the generated covering information systems $(U_i, \mathscr{D}_i)$ for the experiment as follows: (1) the purpose of the experiment is to test the effectiveness of Algorithms 3.6 and 3.13, and it takes a lot of time to transform data sets downloaded from the University of California at Irvine's repository of machine learning databases into covering information systems; (2) the generated covering information systems are suitable for testing the influence of the cardinalities of object sets and attribute sets on computational times; (3) there is one-to-one correspondence between attributes and coverings, and we refer to dynamic covering information systems with variations of attribute sets as dynamic covering information systems with variations of coverings in this section.

4.1. The stability of Algorithms 3.5, 3.6, 3.12 and 3.13

In this section, we illustrate the stability of Algorithms 3.5, 3.6, 3.12 and 3.13 with the experimental results.

Firstly, according to Definition 2.12, we obtain ten sub-covering information systems $\{(U_i, \mathscr{D}^j_i) | j = 1, 2, 3, \ldots, 10\}$ for covering information system $(U_i, \mathscr{D}_i)$ outlined in Table 1, where $\mathscr{D}^j_i = \{\mathscr{C}^i_k | 1 \leq k \leq j \times 10^2\} \subseteq \mathscr{D}_i$. Since there are ten sub-covering information systems whose numbers of objects or cardinalities of covering sets are the same to each other, it is more suitable for testing the influence of the cardinalities of object sets and covering sets on computing approximations of sets.

Secondly, to demonstrate the stability of Algorithms 3.5, 3.6, 3.12, and 3.13, we compute the second and sixth lower and up-

Table 1
Covering information systems for experiments.

No.	Name	$\|U_i\|$	$\|\mathscr{D}_i\|$
1	$(U_1, \mathscr{D}_1)$	2000	1000
2	$(U_2, \mathscr{D}_2)$	4000	1000
3	$(U_3, \mathscr{D}_3)$	6000	1000
4	$(U_4, \mathscr{D}_4)$	8000	1000
5	$(U_5, \mathscr{D}_5)$	10000	1000
6	$(U_6, \mathscr{D}_6)$	12000	1000
7	$(U_7, \mathscr{D}_7)$	14000	1000
8	$(U_8, \mathscr{D}_8)$	16000	1000
9	$(U_9, \mathscr{D}_9)$	18000	1000
10	$(U_{10}, \mathscr{D}_{10})$	20000	1000

Table 2
Computational times using Algorithms NIS, IS, NIX, and IX.

$(U, \mathscr{D}) \setminus t(s)$	Algo.	$\mathscr{D}_i^{1+}$	$\mathscr{D}_i^{2+}$	$\mathscr{D}_i^{3+}$	$\mathscr{D}_i^{4+}$	$\mathscr{D}_i^{5+}$	$\mathscr{D}_i^{6+}$	$\mathscr{D}_i^{7+}$	$\mathscr{D}_i^{8+}$	$\mathscr{D}_i^{9+}$	$\mathscr{D}_i^{10+}$
$(U_1, \mathscr{D}_1)$	NIS	0.2943	0.2907	0.3085	0.3050	0.3028	0.3258	0.3022	0.3151	0.3335	0.3326
	IS	0.0191	0.0187	0.0187	0.0190	0.0189	0.0189	0.0189	0.0188	0.0189	0.0188
	NIX	0.4328	0.4517	0.4696	0.4872	0.5052	0.5153	0.5377	0.5549	0.5634	0.5806
	IX	0.0440	0.0444	0.0445	0.0447	0.0448	0.0462	0.0456	0.0455	0.0452	0.0443
$(U_2, \mathscr{D}_2)$	NIS	1.3922	1.2390	1.3924	1.4876	1.4143	1.3182	1.2613	1.5081	1.4460	1.2898
	IS	0.1161	0.1161	0.1155	0.1148	0.1149	0.1140	0.1141	0.1149	0.1129	0.1139
	NIX	1.7887	1.8439	1.8706	1.8957	1.9566	2.0176	2.0506	2.0670	2.1141	2.1527
	IX	0.2763	0.2765	0.2786	0.2789	0.2769	0.2767	0.2783	0.2782	0.2773	0.2784
$(U_3, \mathscr{D}_3)$	NIS	2.8154	2.7426	2.9496	3.0044	2.9784	3.2760	2.8599	3.0582	3.4272	3.0909
	IS	0.2813	0.2803	0.2819	0.2821	0.2803	0.2812	0.2799	0.2811	0.2804	0.2795
	NIX	4.1230	4.1851	4.2551	4.2700	4.3912	4.4208	4.5532	4.5924	4.5920	4.6849
	IX	0.6934	0.6915	0.6892	0.6915	0.6912	0.6926	0.6917	0.6884	0.6864	0.6873
$(U_4, \mathscr{D}_4)$	NIS	4.8119	4.9112	5.7576	5.2600	5.4411	5.7732	5.5462	5.2157	5.0850	5.6833
	IS	0.5363	0.5341	0.5336	0.5348	0.5345	0.5348	0.5336	0.5334	0.5340	0.5339
	NIX	7.3557	7.4368	7.4743	7.5832	7.7715	7.8551	7.9852	8.0601	8.1282	8.1712
	IX	1.3121	1.3063	1.3066	1.3071	1.3075	1.3100	1.3079	1.3066	1.3076	1.3063
$(U_5, \mathscr{D}_5)$	NIS	7.8162	8.3274	8.7470	8.4818	9.0257	9.5636	9.5964	9.1714	9.0965	8.6726
	IS	0.8992	0.9000	0.9007	0.9014	0.9012	0.8999	0.8993	0.8983	0.9010	0.9007
	NIX	11.6951	11.8407	11.7945	12.0203	12.0654	12.1611	12.3147	12.5509	12.5952	12.8277
	IX	2.2205	2.2174	2.2170	2.2169	2.2171	2.2146	2.2175	2.2150	2.2153	2.2187
$(U_6, \mathscr{D}_6)$	NIS	11.8148	11.9739	11.9246	13.1000	11.8360	11.9592	12.4978	13.3411	13.5243	13.9597
	IS	1.3113	1.3100	1.3097	1.3109	1.3125	1.3140	1.3138	1.3118	1.3131	1.3130
	NIX	16.8548	17.0952	17.2114	17.2539	17.6459	17.9557	17.9717	18.1446	18.2711	18.4869
	IX	3.2322	3.2315	3.2340	3.2328	3.2371	3.2353	3.2337	3.2327	3.2349	3.2352
$(U_7, \mathscr{D}_7)$	NIS	15.3896	17.0839	18.4740	16.9661	17.1482	17.1699	15.8839	19.1676	16.9261	17.4140
	IS	1.8163	1.8119	1.8157	1.8154	1.8156	1.8140	1.8136	1.8158	1.8126	1.8127
	NIX	23.2606	23.3631	23.4501	23.7996	24.1872	24.4196	25.0029	24.7828	25.1679	25.5318
	IX	4.4889	4.4926	4.4869	4.4866	4.4816	4.4913	4.4835	4.4861	4.4932	4.4883
$(U_8, \mathscr{D}_8)$	NIS	19.2680	20.3987	20.4275	25.1255	24.1952	21.6475	22.7576	25.5788	23.0497	24.5230
	IS	2.3293	2.3483	2.3338	2.3395	2.3345	2.3328	2.3310	2.3345	2.3363	2.3286
	NIX	30.7299	31.6841	31.9221	31.8740	32.3828	32.8620	33.1812	33.3335	34.4184	34.2922
	IX	5.7394	5.7299	5.7295	5.7356	5.7260	5.7329	5.7385	5.7348	5.7224	5.7266
$(U_9, \mathscr{D}_9)$	NIS	27.7270	24.6801	27.1033	27.1681	28.1967	29.5912	27.2404	28.7746	30.5663	33.0124
	IS	3.0808	3.0785	3.0767	3.0843	3.0800	3.0772	3.0818	3.0821	3.0768	3.0762
	NIX	38.3136	40.5511	40.2451	40.7515	41.2912	41.8698	42.1964	42.3846	42.2559	42.9560
	IX	7.6318	7.6339	7.6263	7.6390	7.6344	7.6334	7.6343	7.6320	7.6321	7.6454
$(U_{10}, \mathscr{D}_{10})$	NIS	38.7478	31.2401	35.1207	35.7740	37.2349	36.7227	38.3850	37.1738	40.7795	36.0968
	IS	3.7285	3.7297	3.7251	3.7269	3.7190	3.7233	3.7238	3.7213	3.7216	3.7215
	NIX	47.8739	49.2954	50.3553	52.0011	50.0680	53.7648	52.6960	54.1776	53.6499	55.9171
	IX	9.2187	9.2226	9.2298	9.2168	9.2150	9.2176	9.2193	9.2254	9.2165	9.2202

per approximations of sets in sub-covering information systems $\{(U_i, \mathscr{D}_i^j)|i, j = 1, 2, 3, \ldots, 10\}$. For example, we show the process of computing the second and sixth lower and upper approximations of sets in covering information system $(U_1, \mathscr{D}_1^1)$, where $|U_1| = 2000$ and $|\mathscr{D}_1^1| = 100$ as follows. By adding a covering into $\mathscr{D}_1^1$, we obtain the dynamic covering information system $(U_1, \mathscr{D}_1^{1+})$, where $|U_1| = 2000$ and $|\mathscr{D}_1^{1+}| = 101$. Taking any $X \subseteq U_1$, we compute the second lower and upper approximations of X in dynamic covering information system $(U_1, \mathscr{D}_1^{1+})$ using Algorithms 3.5 and 3.6. Furthermore, we compute the sixth lower and upper approximations of X in the dynamic covering information system $(U_1, \mathscr{D}_1^{1+})$ using Algorithms 3.12 and 3.13. To confirm the accuracy of the experiment results, we conduct each experiment ten times and show the average time of ten experimental results in Table 2, where $t(s)$ denotes that the measure of time is in seconds. Concretely, from the third column of Table 2, we see that 0.2943, 0.4328, 0.0191, and 0.0440 are the average times of computing approximations of sets using Algorithms 3.5, 3.6, 3.12, and 3.13, respectively, in the dynamic covering information system $(U_1, \mathscr{D}_1^{1+})$. Especially, we see Algorithms 3.6 and 3.13 are more effective than Algorithms 3.5 and 3.12, respectively, in $(U_1, \mathscr{D}_1^{1+})$. From the third column of Table 3, we also get the standard deviations 0.0050, 0.0045, 0.0010, and 0.0004 of ten computational times using Algorithms 3.5, 3.6, 3.12, and 3.13, respectively, in the dynamic covering information system $(U_1, \mathscr{D}_1^{1+})$. We see Algorithms 3.6 and 3.13 are more stable than Algorithms 3.5 and 3.12, respectively, for computing the second and sixth lower and upper approximations of sets in the dynamic covering information system $(U_1, \mathscr{D}_1^{1+})$.

Thirdly, we compute the second and sixth lower and upper approximations of sets similarly in dynamic covering information systems $\{(U_i, \mathscr{D}_i^{j+})|i, j = 1, 2, 3, \ldots, 10\}$ and show the average times and standard deviations in Tables 2 and 3, respectively. The standard deviations in Tables 3 illustrate Algorithms 3.5, 3.6, 3.12, and 3.13 are stable to compute the second and sixth lower and upper approximations of sets in dynamic covering information systems $\{(U_i, \mathscr{D}_i^{j+})|i, j = 1, 2, 3, \ldots, 10\}$. Especially, Algorithms 3.6 and 3.13 are more stable to compute the second and sixth lower and upper approximations of sets than Algorithms 3.5 and 3.12, respectively, in dynamic covering information systems. Furthermore, we show the standard deviations of computational times using Algorithms 3.5, 3.6, 3.12, and 3.13 by Fig. 1, and see the curve lines of standard deviations using Algorithms 3.5 and 3.12 fluctuate above those of Algorithms 3.6 and 3.13, respectively. Therefore, Algorithms 3.6 and 3.13 are more stable than Algorithms 3.5 and 3.12 for computing approximations of sets in dynamic covering information systems, respectively.

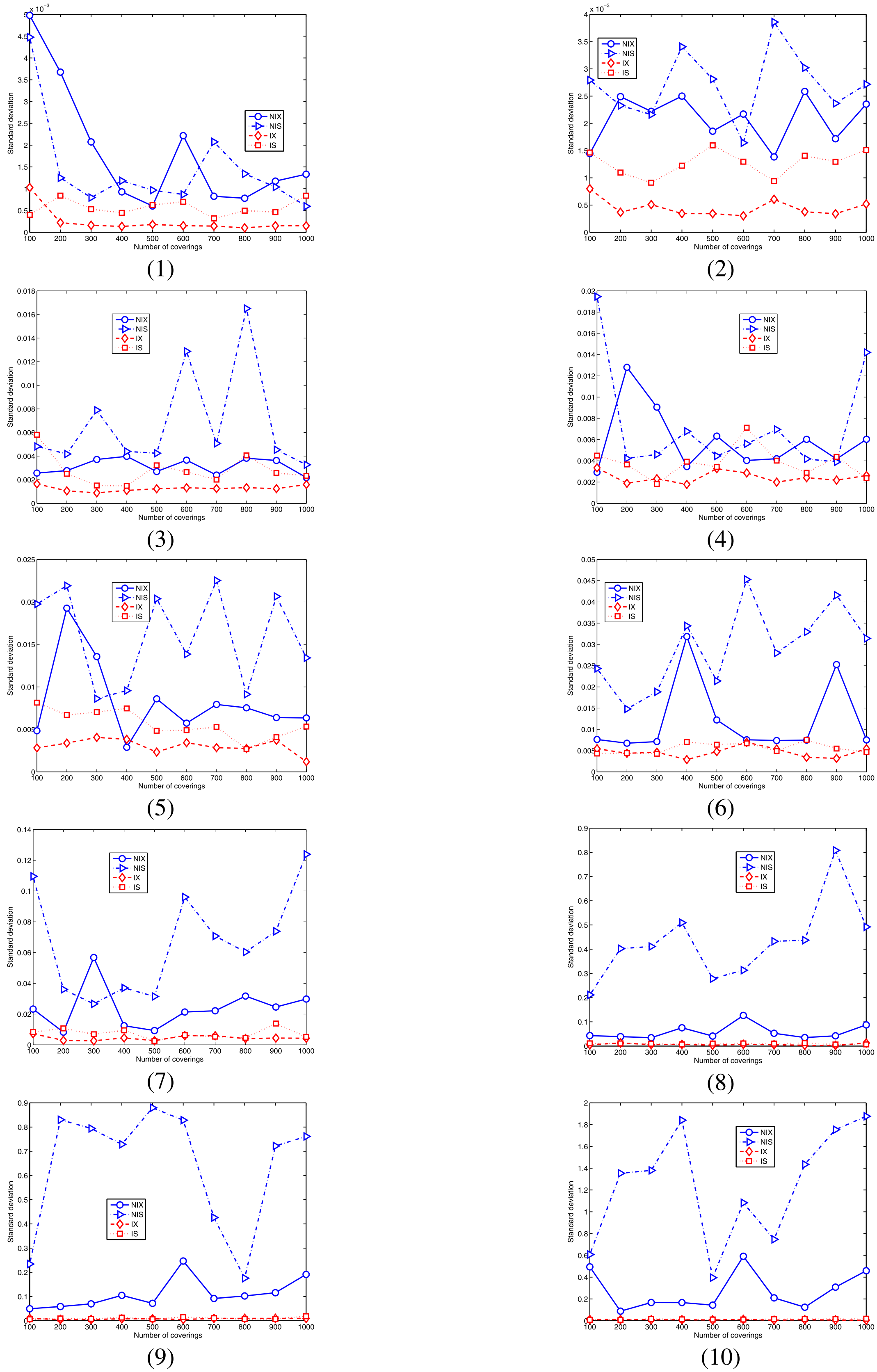

Fig. 1. Standard deviations of computational times using Algorithms NIS, IS, NIX, and IX.

Table 3
Standard deviations of computational times using Algorithms NIS, IS, NIX, and IX.

$(U, \mathscr{D})$	Algo.	$\mathscr{D}_i^{1+}$	$\mathscr{D}_i^{2+}$	$\mathscr{D}_i^{3+}$	$\mathscr{D}_i^{4+}$	$\mathscr{D}_i^{5+}$	$\mathscr{D}_i^{6+}$	$\mathscr{D}_i^{7+}$	$\mathscr{D}_i^{8+}$	$\mathscr{D}_i^{9+}$	$\mathscr{D}_i^{10+}$
$(U_1, \mathscr{D}_1)$	NIS	0.0050	0.0037	0.0021	0.0009	0.0006	0.0022	0.0008	0.0008	0.0012	0.0013
	IS	0.0010	0.0002	0.0002	0.0001	0.0002	0.0002	0.0001	0.0001	0.0001	0.0001
	NIX	0.0045	0.0012	0.0008	0.0012	0.0010	0.0009	0.0021	0.0013	0.0010	0.0006
	IX	0.0004	0.0008	0.0005	0.0004	0.0006	0.0007	0.0003	0.0005	0.0005	0.0008
$(U_2, \mathscr{D}_2)$	NIS	0.0014	0.0025	0.0022	0.0025	0.0019	0.0022	0.0014	0.0026	0.0017	0.0024
	IS	0.0008	0.0004	0.0005	0.0003	0.0003	0.0003	0.0006	0.0004	0.0003	0.0005
	NIX	0.0028	0.0023	0.0022	0.0034	0.0028	0.0016	0.0039	0.0030	0.0024	0.0027
	IX	0.0015	0.0011	0.0009	0.0012	0.0016	0.0013	0.0009	0.0014	0.0013	0.0015
$(U_3, \mathscr{D}_3)$	NIS	0.0026	0.0028	0.0037	0.0040	0.0027	0.0037	0.0024	0.0038	0.0036	0.0022
	IS	0.0016	0.0011	0.0009	0.0011	0.0012	0.0013	0.0012	0.0013	0.0012	0.0016
	NIX	0.0048	0.0042	0.0079	0.0044	0.0042	0.0129	0.0051	0.0165	0.0045	0.0033
	IX	0.0058	0.0025	0.0015	0.0015	0.0032	0.0027	0.0020	0.0041	0.0026	0.0023
$(U_4, \mathscr{D}_4)$	NIS	0.0029	0.0128	0.0091	0.0035	0.0063	0.0040	0.0042	0.0060	0.0042	0.0060
	IS	0.0033	0.0019	0.0023	0.0018	0.0033	0.0029	0.0020	0.0024	0.0022	0.0026
	NIX	0.0195	0.0042	0.0046	0.0068	0.0045	0.0056	0.0070	0.0042	0.0039	0.0142
	IX	0.0045	0.0037	0.0018	0.0039	0.0034	0.0071	0.0040	0.0029	0.0044	0.0024
$(U_5, \mathscr{D}_5)$	NIS	0.0048	0.0193	0.0136	0.0029	0.0086	0.0057	0.0079	0.0075	0.0064	0.0063
	IS	0.0028	0.0034	0.0041	0.0038	0.0023	0.0034	0.0028	0.0028	0.0037	0.0012
	NIX	0.0198	0.0219	0.0086	0.0096	0.0204	0.0139	0.0225	0.0091	0.0206	0.0134
	IX	0.0081	0.0067	0.0070	0.0075	0.0048	0.0049	0.0053	0.0027	0.0041	0.0053
$(U_6, \mathscr{D}_6)$	NIS	0.0076	0.0067	0.0071	0.0319	0.0122	0.0076	0.0074	0.0075	0.0253	0.0075
	IS	0.0054	0.0044	0.0046	0.0029	0.0048	0.0070	0.0054	0.0034	0.0032	0.0055
	NIX	0.0243	0.0148	0.0188	0.0344	0.0214	0.0453	0.0280	0.0329	0.0416	0.0314
	IX	0.0043	0.0045	0.0043	0.0070	0.0064	0.0067	0.0049	0.0075	0.0055	0.0047
$(U_7, \mathscr{D}_7)$	NIS	0.0233	0.0082	0.0568	0.0124	0.0093	0.0214	0.0221	0.0317	0.0246	0.0298
	IS	0.0072	0.0029	0.0026	0.0045	0.0027	0.0060	0.0059	0.0041	0.0044	0.0043
	NIX	0.1095	0.0360	0.0267	0.0370	0.0314	0.0961	0.0707	0.0604	0.0738	0.1239
	IX	0.0083	0.0107	0.0069	0.0095	0.0026	0.0063	0.0053	0.0046	0.0139	0.0052
$(U_8, \mathscr{D}_8)$	NIS	0.0431	0.0389	0.0344	0.0754	0.0412	0.1264	0.0528	0.0350	0.0423	0.0873
	IS	0.0052	0.0126	0.0058	0.0071	0.0036	0.0070	0.0057	0.0032	0.0025	0.0125
	NIX	0.2113	0.4023	0.4112	0.5095	0.2784	0.3134	0.4325	0.4373	0.8081	0.4922
	IX	0.0110	0.0107	0.0096	0.0066	0.0104	0.0108	0.0101	0.0123	0.0063	0.0069
$(U_9, \mathscr{D}_9)$	NIS	0.0496	0.0586	0.0697	0.1048	0.0720	0.2466	0.0919	0.1022	0.1157	0.1913
	IS	0.0096	0.0051	0.0050	0.0079	0.0082	0.0053	0.0096	0.0098	0.0097	0.0098
	NIX	0.2342	0.8303	0.7943	0.7288	0.8797	0.8283	0.4264	0.1759	0.7218	0.7613
	IX	0.0060	0.0093	0.0077	0.0129	0.0063	0.0150	0.0107	0.0074	0.0074	0.0186
$(U_{10}, \mathscr{D}_{10})$	NIS	0.4949	0.0878	0.1672	0.1667	0.1434	0.5916	0.2096	0.1244	0.3082	0.4589
	IS	0.0105	0.0113	0.0086	0.0085	0.0074	0.0117	0.0052	0.0093	0.0066	0.0048
	NIX	0.6078	1.3534	1.3801	1.8408	0.3955	1.0828	0.7471	1.4347	1.7540	1.8770
	IX	0.0076	0.0079	0.0170	0.0128	0.0113	0.0109	0.0148	0.0165	0.0158	0.0181

4.2. The influence of the number of objects

In this section, we analyze the influence of the number of objects on time of computing the second and sixth lower and upper approximations of sets in dynamic covering information systems.

Firstly, we compare the times of computing the second lower and upper approximations of sets using Algorithms 3.5 with those using Algorithm 3.6 in dynamic covering information systems with the same cardinality of covering sets. From the results in Table 2, we see that the computing times are increasing with the increasing number of objects using Algorithms 3.5 and 3.6. For example, from the third column of Table 2, we have the runtimes {0.2943, 1.3922, 2.81540, 4.8119, 7.8162, 11.8148, 15.3896, 19.2680, 27.7270, 38.7478} and {0.0191, 0.1161, 0.2813, 0.5363, 0.8992, 1.3113, 1.8163, 2.32930, 3.0808, 3.7285} with Algorithms 3.5 and 3.6, respectively, in dynamic covering information systems $\{(U_i, \mathscr{D}_i^{1+}) \mid 1 \leq i \leq 10\}$. We also find that Algorithm 3.6 executes faster than Algorithms 3.5 in dynamic covering information systems. For example, from the above results, we have that $0.0191 < 0.2943$, $0.1161 < 1.3922$, $0.2813 < 2.8154$, $0.5363 < 4.8119$, $0.8992 < 7.8162$, $1.3113 < 11.8148$, $1.8163 < 15.3896$, $2.3293 < 19.2680$, $3.0808 < 27.7270$, and $3.7285 < 38.7478$.

Secondly, we compare the times of computing the sixth lower and upper approximations of sets using Algorithms 3.12 with those using Algorithms 3.13 in dynamic covering information systems with the same cardinality of covering sets. From the results in Table 2, we see that the computing times are increasing with the increasing cardinalities of object sets using Algorithms 3.12 and 3.13. For example, from the third column of Table 2, we have the runtimes {0.4328, 1.7887, 4.1230, 7.3557, 11.6951, 16.8548, 23.2606, 30.7299, 38.3136, 47.8739} and {0.0440, 0.2763, 0.6934, 1.3121, 2.2205, 3.2322, 4.4889, 5.7394, 7.6318, 9.2187} with Algorithms 3.12 and 3.13, respectively, in dynamic covering information systems $\{(U_i, \mathscr{D}_i^{1+}) \mid 1 \leq i \leq 10\}$. We also find that Algorithms 3.13 executes faster than Algorithms 3.12 in dynamic covering information systems. For example, from the above results, we have that $0.0440 < 0.4328$, $0.2763 < 1.7887$, $0.6934 < 4.1230$, $1.3121 < 7.3557$, $2.2205 < 11.6951$, $3.2322 < 16.8548$, $4.4889 < 23.2606$, $5.7394 < 30.7299$, $7.6318 < 38.3136$, and $9.2187 < 47.8739$.

Thirdly, we show these results in Fig. 2 to illustrate the effectiveness of Algorithms 3.6 and 3.13. For example, Fig. 2(j) illustrates the times of computing the second and sixth lower and upper approximations of sets using Algorithms 3.5, 3.6, 3.12, and 3.13 in dynamic covering information systems $\{(U_i, \mathscr{D}_i^{j+}) | 1 \leq i \leq 10\}$. In each figure, *NIS, IS, NIX*, and *IX* mean Algorithms 3.5, 3.6, 3.12, and 3.13, respectively; i stands for the cardinality of object set in X Axis, while the y-coordinate stands for the time to construct the approximations of concepts. Therefore, Algorithms 3.6 and 3.13 are effective to compute the second and sixth lower and upper approxi-

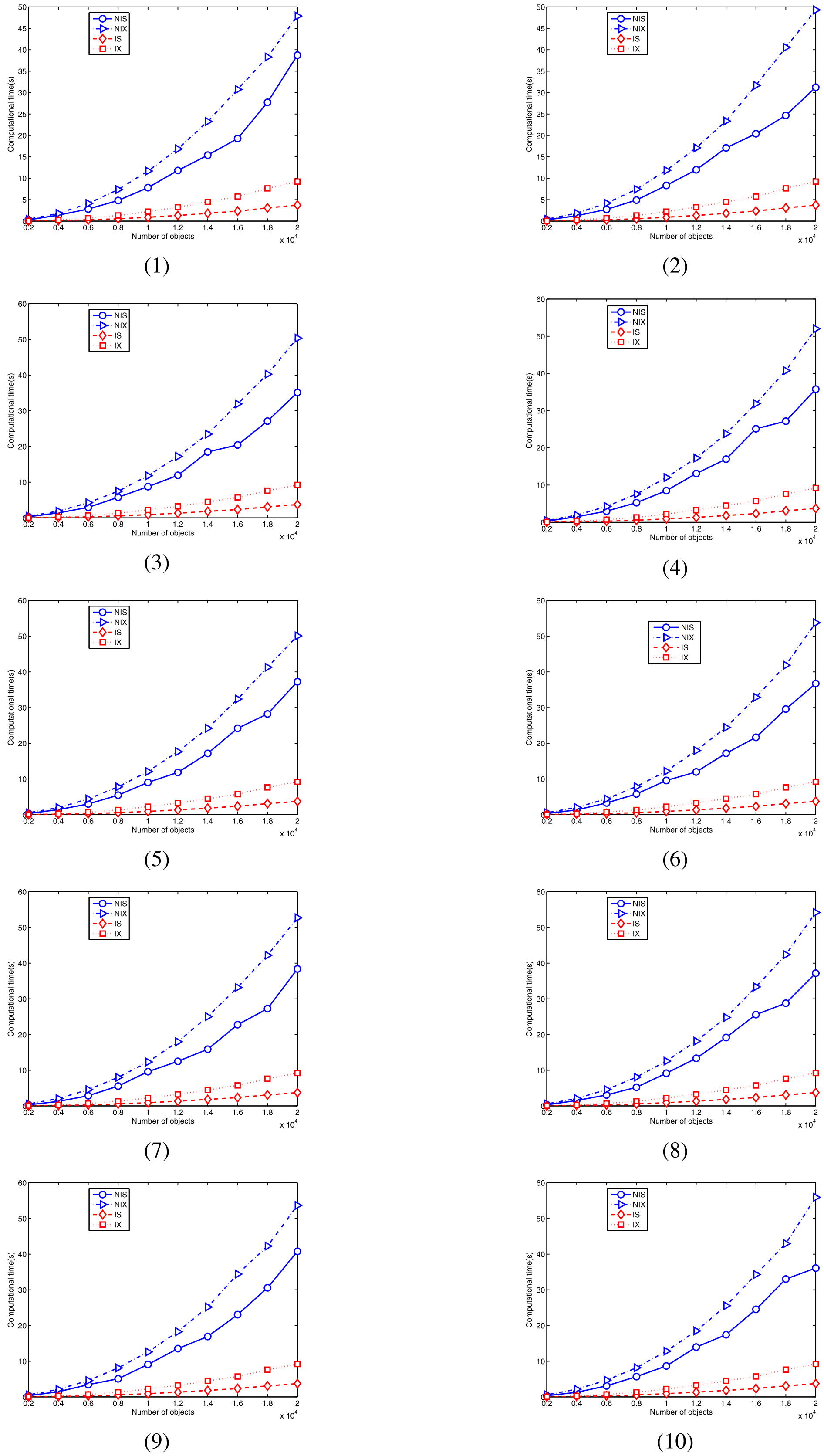

Fig. 2. Computational times using Algorithms NIS, IS, NIX, and IX in $\{(U_i, \mathscr{D}_i^{j+})|1 \leq i \leq 10\}(1 \leq j \leq 10)$.

mations of sets, respectively, in dynamic covering information systems.

4.3. The influence of the cardinality of covering set

In this section, we analyze the influence of the cardinality of covering set on time of computing the second and sixth lower and upper approximations of sets in dynamic covering information systems.

Firstly, we compare the times of computing the second lower and upper approximations of sets using Algorithms 3.5 with those using Algorithm 3.6 in dynamic covering information systems with the same number of objects. According to the results in Table 2, we see that the computing times are almost not increasing with the increasing cardinalities of covering sets using Algorithms 3.5 and 3.6. For example, from the second row of Table 2, we have the runtimes {0.2943, 0.2907, 0.3085, 0.3050, 0.3028, 0.3258, 0.3022, 0.3151, 0.3335, 0.3326} and {0.0191, 0.0187, 0.0187, 0.0190, 0.0189, 0.0189, 0.0189, 0.0188, 0.0189, 0.0188} with Algorithms 3.5 and 3.6, respectively, in dynamic covering information systems $\{(U_1, \mathscr{D}_1^{j+}) \mid 1 \leq j \leq 10\}$. We also find that Algorithm 3.6 executes faster than Algorithms 3.5 in dynamic covering information systems. For example, from the above results, we have that $0.0191 < 0.2943$, $0.0187 < 0.2907$, $0.0187 < 0.3085$, $0.0190 < 0.3050$, $0.0189 < 0.3028$, $0.0189 < 0.3258$, $0.0189 < 0.3022$, $0.0188 < 0.3151$, $0.0189 < 0.3335$ and $0.0188 < 0.3326$.

Secondly, we compare the times of computing the sixth lower and upper approximations of sets using Algorithms 3.12 with those using Algorithms 3.13 in dynamic covering information systems with the same number of objects. From the results in Table 2, we see that the computing times are increasing with the increasing cardinalities of covering sets using Algorithms 3.12. For example, from the second row of Table 2, we have the runtimes {0.4328, 0.4517, 0.4696, 0.4872, 0.5052, 0.5153, 0.5377, 0.5549, 0.5634, 0.5806} and {0.0440, 0.0444, 0.0445, 0.0447, 0.0448, 0.0462, 0.0456, 0.0455, 0.0452, 0.0443} with Algorithms 3.12 and 3.13 in dynamic covering information systems $\{(U_1, \mathscr{D}_1^{j+}) \mid 1 \leq j \leq 10\}$. But the computing times are almost not increasing with the increasing cardinalities of covering sets using Algorithms 3.13. We also find that Algorithms 3.13 executes faster than Algorithms 3.12 in dynamic covering information systems. For example, from the above results, we have that $0.0440 < 0.4328$, $0.0444 < 0.4517$, $0.0445 < 0.4696$, $0.0447 < 0.4872$, $0.0448 < 0.5052$, $0.0462 < 0.5153$, $0.0456 < 0.5377$, $0.0455 < 0.5549$, $0.0452 < 0.5634$ and $0.0443 < 0.5806$.

Thirdly, we show these results in Fig. 3 to illustrate the effectiveness of Algorithms 3.6 and 3.13. For example, Fig. 3(i) illustrates the times of computing the second and sixth lower and upper approximations of sets using Algorithms 3.5, 3.6, 3.12, and 3.13 in dynamic covering information systems $\{(U_i, \mathscr{D}_i^{j+}) | 1 \leq i \leq 10\}$. In each figure, *NIS*, *IS*, *NIX*, and *IX* mean Algorithms 3.5, 3.6, 3.12, and 3.13, respectively; i stands for the cardinality of covering set in X Axis, while the y-coordinate stands for the time to construct the approximations of concepts. Therefore, Algorithms 3.6 and 3.13 are effective to compute the second and sixth lower and upper approximations of sets, respectively, in dynamic covering information systems.

5. Knowledge reduction of dynamic covering information systems

In this section, we employ examples to illustrate how to conduct knowledge reduction of dynamic covering information systems.

Example 5.1. Let $(U, \mathscr{D}_C \cup \mathscr{D}_D)$ be a covering decision information system, where $\mathscr{D}_C = \{\mathscr{C}_1, \mathscr{C}_2, \mathscr{C}_3\}$, $\mathscr{C}_1 = \{\{x_1, x_2, x_3, x_4\}, \{x_5\}\}$, $\mathscr{C}_2 = \{\{x_1, x_2\}, \{x_3, x_4, x_5\}\}$, $\mathscr{C}_3 = \{\{x_1, x_2, x_5\}, \{x_3, x_4\}\}$, $\mathscr{D}_D = \{D_1, D_2\}$, $D_1 = \{x_1, x_2\}$, and $D_2 = \{x_3, x_4, x_5\}$. Firstly, according to Definition 2.8, we obtain

$$\Gamma(\mathscr{D}_C) = M_{\mathscr{D}_C} \bullet M_{\mathscr{D}_C}^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Secondly, by Definition 2.9, we have

$$\Gamma(\mathscr{D}_C) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{D}_C) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Thirdly, according to Definition 2.8, we get

$$\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_1) = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_3) = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Fourthly, by Definition 2.9, we derive

$$\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix},$$

$$\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

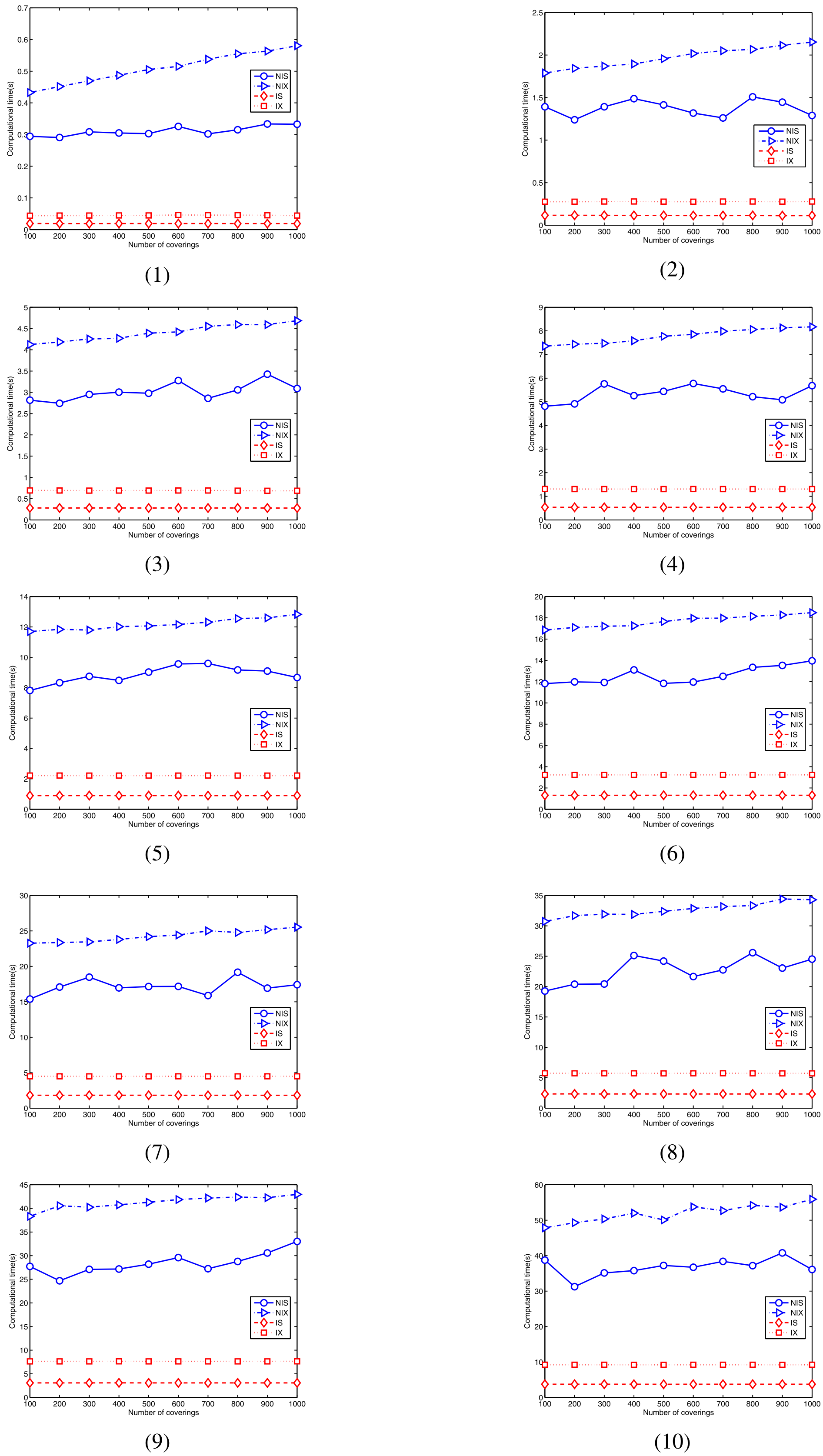

Fig. 3. Computational times using Algorithms NIS, IS, NIX, and IX in $\{(U_i, \mathscr{D}_i^{j+})|1 \leq j \leq 10\}(1 \leq i \leq 10)$.

$$= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_1) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_1) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_3) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_3) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

Therefore, according to Definition 2.9, $\{\mathscr{C}_1, \mathscr{C}_3\}$ is a type-1 reduct of $(U, \mathscr{D}_C \cup \mathscr{D}_D)$.

In Example 5.1, we must compute $\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \bullet M_{\mathscr{D}_D}$, $\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \odot M_{\mathscr{D}_D}, \Gamma(\mathscr{C}_1) \bullet M_{\mathscr{D}_D}, \Gamma(\mathscr{C}_1) \odot M_{\mathscr{D}_D}, \Gamma(\mathscr{C}_3) \bullet M_{\mathscr{D}_D}$, and $\Gamma(\mathscr{C}_3) \odot M_{\mathscr{D}_D}$ for constructing type-1 reducts of covering decision information system $(U, \mathscr{D}_C \cup \mathscr{D}_D)$.

In what follows, we employ an example to illustrate how to compute type-1 reducts of dynamic covering decision information systems with the immigration of attributes.

Example 5.2. (Continued from Example 5.1) Let $(U, \mathscr{D}_C^+ \cup \mathscr{D}_D)$ be a dynamic covering decision information system of $(U, \mathscr{D}_C \cup \mathscr{D}_D)$, where $\mathscr{D}_C^+ = \{\mathscr{C}_1, \mathscr{C}_2, \mathscr{C}_3, \mathscr{C}_4\}$, $\mathscr{C}_1 = \{\{x_1, x_2, x_3, x_4\}, \{x_5\}\}$, $\mathscr{C}_2 = \{\{x_1, x_2\}, \{x_3, x_4, x_5\}\}$, $\mathscr{C}_3 = \{\{x_1, x_2, x_5\}, \{x_3, x_4\}\}$, $\mathscr{C}_4 = \{\{x_1, x_2\}, \{x_3, x_4\}, \{x_5\}\}$, $\mathscr{D}_D = \{D_1, D_2\}$, $D_1 = \{x_1, x_2\}$, and $D_2 = \{x_3, x_4, x_5\}$. Firstly, by Theorem 3.4 and Example 5.1, we obtain

$$\Gamma(\mathscr{D}_C^+) = \Gamma(\mathscr{D}_C) \bigvee \Gamma(\mathscr{C}_4) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Secondly, by Definition 2.9, we have

$$\Gamma(\mathscr{D}_C^+) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{D}_C^+) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Thirdly, by Example 5.1, we get

$$\Gamma(\mathscr{D}_C) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}, \Gamma(\mathscr{D}_C) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}, \Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_1) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}, \Gamma(\mathscr{C}_1) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_3) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}, \Gamma(\mathscr{C}_3) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

Therefore, according to Definition 2.10, $\{\mathscr{C}_1, \mathscr{C}_3\}$ is a type-1 reduct of $(U, \mathscr{D}_C^+ \cup \mathscr{D}_D)$.

In Example 5.2, we must compute $\Gamma(\mathscr{D}_C^+) \bullet M_{\mathscr{D}_D}$, $\Gamma(\mathscr{D}_C^+) \odot M_{\mathscr{D}_D}, \Gamma(\mathscr{D}_C) \bullet M_{\mathscr{D}_D}, \Gamma(\mathscr{D}_C) \odot M_{\mathscr{D}_D}, \Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \bullet M_{\mathscr{D}_D}, \Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \odot M_{\mathscr{D}_D}, \Gamma(\mathscr{C}_1) \bullet M_{\mathscr{D}_D}, \Gamma(\mathscr{C}_1) \odot M_{\mathscr{D}_D}, \Gamma(\mathscr{C}_3) \bullet M_{\mathscr{D}_D}$, and $\Gamma(\mathscr{C}_3) \odot M_{\mathscr{D}_D}$ if we construct type-1 reducts of dynamic covering decision information system $(U, \mathscr{D}_C^+ \cup \mathscr{D}_D)$ with non-incremental approach. But we only need to compute $\Gamma(\mathscr{D}_C^+) \bullet M_{\mathscr{D}_D}$ and $\Gamma(\mathscr{D}_C^+) \odot M_{\mathscr{D}_D}$ for constructing type-1 reducts of $(U, \mathscr{D}_C^+ \cup \mathscr{D}_D)$ with incremental approach. Moreover, we similarly construct type-2 reducts of dynamic covering decision information systems with the immigration of attributes. Therefore, the designed algorithm is effective to conduct knowledge reduction of dynamic covering decision information systems with the immigration of attributes.

We employ another example to illustrate how to compute type-1 reducts of dynamic covering decision information systems with the emigration of attributes.

Example 5.3. (Continued from Example 5.1) Let $(U, \mathscr{D}_C^- \cup \mathscr{D}_D)$ be a dynamic covering decision information system of $(U, \mathscr{D}_C \cup \mathscr{D}_D)$, where $\mathscr{D}_C^- = \{\mathscr{C}_1, \mathscr{C}_3\}$, $\mathscr{C}_1 = \{\{x_1, x_2, x_3, x_4\}, \{x_5\}\}$, $\mathscr{C}_3 = \{\{x_1, x_2, x_5\}, \{x_3, x_4\}\}$, $\mathscr{D}_D = \{D_1, D_2\}$, $D_1 = \{x_1, x_2\}$, and $D_2 = \{x_3, x_4, x_5\}$. By Example 5.1, we obtain

$$\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}, \Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_1) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}, \Gamma(\mathscr{C}_1) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix},$$

$$\Gamma(\mathscr{C}_3) \bullet M_{\mathscr{D}_D} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}, \Gamma(\mathscr{C}_3) \odot M_{\mathscr{D}_D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

Therefore, according to Definition 2.10, $\{\mathscr{C}_1, \mathscr{C}_3\}$ is a type-1 reduct of $(U, \mathscr{D}_C^- \cup \mathscr{D}_D)$.

In Example 5.3, we must compute $\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \bullet M_{\mathscr{D}_D}$, $\Gamma(\{\mathscr{C}_1, \mathscr{C}_3\}) \odot M_{\mathscr{D}_D}$, $\Gamma(\mathscr{C}_1) \bullet M_{\mathscr{D}_D}$, $\Gamma(\mathscr{C}_1) \odot M_{\mathscr{D}_D}$, $\Gamma(\mathscr{C}_3) \bullet M_{\mathscr{D}_D}$, and $\Gamma(\mathscr{C}_3) \odot M_{\mathscr{D}_D}$ if we construct type-1 reducts of dynamic covering decision information system $(U, \mathscr{D}_C^- \cup \mathscr{D}_D)$ with non-incremental approach. But we construct type-1 reducts of $(U, \mathscr{D}_C^- \cup \mathscr{D}_D)$ without computation using incremental approach. Furthermore, we similarly construct type-2 reducts of dynamic covering decision information systems with the emigration of attributes. Therefore, the incremental algorithm is effective to conduct knowledge reduction of dynamic covering decision information systems with the emigration of attributes.

6. Conclusions

In this paper, we have updated the type-1 and type-2 characteristic matrices with incremental learning technique and designed effective algorithms for computing the second and sixth lower and upper approximations of sets in dynamic covering information systems with variations of attributes. We have explored several examples to illustrate how to calculate the second and sixth lower and upper approximations of sets. We have employed experimental results to illustrate the designed algorithms are effective to calculate the second and sixth lower and upper approximations of sets in dynamic covering information systems with the immigration of attributes. We have explored several examples to demonstrate how to conduct knowledge reduction of dynamic covering decision information systems.

In the future, we will investigate further the calculation of approximations of sets in dynamic covering information systems and propose effective algorithms for attribute reduction of dynamic covering information systems. Furthermore, we will provide parallel algorithms for knowledge reduction of dynamic covering information systems from the view of matrices.

Acknowledgments

We would like to thank the anonymous reviewers very much for their professional comments and valuable suggestions. This work is supported by the National Natural Science Foundation of China (No. 61673301, 61603063, 11526039, 61573255), Specialized Research Fund for the Doctoral Program of Higher Education of China (No. 20130072130004), China Postdoctoral Science Foundation (No. 2015M580353), China Postdoctoral Science Special Foundation (No. 2016T90383), the Scientific Research Fund of Hunan Provincial Education Department (No. 15B004).

References

[1] N. Azam, J.T. Yao, Interpretation of equilibria in game-theoretic rough sets, Inf. Sci. 295 (2015) 586–599.
[2] M.J. Cai, Q.G. Li, J.M. Ma, Knowledge reduction of dynamic covering decision information systems caused by variations of attribute values, Int. J. Mach. Learn. Cybern. (2017), doi:10.1007/s13042-015-0484-9.
[3] 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 Trans. Knowl. Data Eng. 25 (2) (2013) 174–184.
[4] D.G. Chen, Y.Y. Yang, Z. Dong, An incremental algorithm for attribute reduction with variable precision rough sets, Appl. Soft Comput. 45 (2016) 129–149.
[5] D.G. Chen, X.X. Zhang, W.L. Li, On measurements of covering rough sets based on granules and evidence theory, Inf. Sci. 317 (2015) 329–348.
[6] Y.N. Fan, T.L. Tseng, C.C. Chern, C.C. Huang, Rule induction based on an incremental rough set, Expert Syst. Appl. 36 (2009) 11439–11450.
[7] Q.R. Feng, D.Q. Miao, Y. Cheng, Hierarchical decision rules mining, Expert Syst. Appl. 37 (2010) 2081–2091.
[8] S. Foithong, O. Pinngern, B. Attachoo, Feature subset selection wrapper based on mutual information and rough sets, Expert Syst. Appl. 39 (2012) 574–584.
[9] J. Hu, T.R. Li, C. Luo, H. Fujita, S.Y. Li, Incremental fuzzy probabilistic rough sets over two universes, Int. J. Approximate Reasoning 81 (2017) 28–48.
[10] C.X. Hu, S.X. Liu, G.X. Liu, Matrix-based approaches for dynamic updating approximations in multigranulation rough sets, Knowl. Based Syst. 122 (2017) 51–63.
[11] Q.H. Hu, D.R. Yu, Z.X. Xie, Neighborhood classifiers, Expert Syst. Appl. 34 (2008) 866–876.
[12] B. Huang, C.X. Guo, H.X. Li, G.F. Feng, X.Z. Zhou, An intuitionistic fuzzy graded covering rough set, Knowl. Based Syst. 107 (2016) 155–178.
[13] 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 (2017a) 273–283.
[14] 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 (2017b) 131–147.
[15] G.M. Lang, Q.G. Li, M.J. Cai, T. Yang, Characteristic matrices-based knowledge reduction in dynamic covering decision information systems, Knowl. Based Syst. 85 (2015) 1–26.
[16] G.M. Lang, Q.G. Li, M.J. Cai, T. Yang, Q.M. Xiao, Incremental approaches to constructing approximations of sets based on characteristic matrices, Int. J. Mach. Learn. Cybern. 8 (2017) 203–222.
[17] G.M. Lang, D.Q. Miao, M.J. Cai, Three-way decision approaches to conflict analysis using decision-theoretic rough set theory, Inf. Sci. 406–407 (2017) 185–207.
[18] G.M. Lang, D.Q. Miao, T. Yang, M.J. Cai, Knowledge reduction of dynamic covering decision information systems when varying covering cardinalities, Inf. Sci. 346–347 (2016) 236–260.
[19] Y. Leung, W.Z. Wu, W.X. Zhang, Knowledge acquisition in incomplete information systems: a rough set approach, Eur. J. Oper. Res. 168 (2006) 164–180.
[20] S.Y. Li, T.R. Li, D. Liu, Incremental updating approximations in dominance-based rough sets approach under the variation of the attribute set, Knowl. Based Syst. 40 (2013) 17–26.
[21] 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, Int. J. Intell. Syst. 28 (8) (2013) 729–751.
[22] 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 (5) (2007) 485–494.
[23] T.R. Li, D. Ruan, J. Song, Dynamic maintenance of decision rules with rough set under characteristic relation, Wireless Communications, Networking and Mobile Computing (2007) 3713–3716.
[24] J.H. Li, C.L. Mei, Y.J. Lv, Incomplete decision contexts: approximate concept construction, rule acquisition and knowledge reduction, Int. J. Approximate Reasoning 54 (1) (2013) 149–165.
[25] Y. Li, Z.H. Zhang, W.B. Chen, F. Min, TDUP: An approach to incremental mining of frequent itemsets with three-way-decision pattern updating, Int. J. Mach. Learn. Cybern. 8 (2) (2017) 441–453.
[26] J.Y. Liang, F. Wang, C.Y. Dang, Y.H. Qian, A group incremental approach to feature selection applying rough set technique, IEEE Trans. Knowl. Data Eng. 26 (2) (2014) 294–308.
[27] G.L. Liu, The axiomatization of the rough set upper approximation operations, Fundam. Inf. 69 (3) (2006) 331–342.
[28] G.L. Liu, Special types of coverings and axiomatization of rough sets based on partial orders, Knowl. Based Syst. 85 (2015) 316–321.
[29] 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.
[30] D. Liu, D.C. Liang, C.C. Wang, A novel three-way decision model based on incomplete information system, Knowl. Based Syst. 91 (2016) 32–45.
[31] C.H. Liu, D.Q. Miao, J. Qian, On multi-granulation covering rough sets, Int. J. Approximate Reasoning 55 (6) (2014) 1404–1418.
[32] 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.
[33] C. Luo, T.R. Li, H.M. Chen, H. Fujita, Y. Zhang, Efficient updating of probabilistic approximations with incremental objects, Knowl. Based Syst. 109 (2016) 71–83.
[34] L.W. Ma, Two fuzzy covering rough set models and their generalizations over fuzzy lattices, Fuzzy Sets Syst. 294 (2016) 1–17.
[35] D.Q. Miao, C. Gao, N. Zhang, Z.F. Zhang, Diverse reduct subspaces based co–training for partially labeled data, Int. J. Approximate Reasoning 52 (8) (2011) 1103–1117.
[36] J. Qian, C.Y. Dang, X.D. Yue, N. Zhang, Attribute reduction for sequential three-way decisions under dynamic granulation, Int. J. Approximate Reasoning 85 (2017) 196–216.
[37] Y.H. Qian, J.Y. Liang, D.Y. Li, F. Wang, N.N. Ma, Approximation reduction in inconsistent incomplete decision tables, Knowl. Based Syst. 23 (5) (2010) 427–433.
[38] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (5) (1982) 341–356.
[39] G. Peters, S. Poon, Analyzing IT business values-a dominance based rough sets approach perspective, Expert Syst. Appl. 38 (2011) 11120–11128.
[40] Y.L. Sang, J.Y. Liang, Y.H. Qian, Decision-theoretic rough sets under dynamic granulation, Knowl. Based Syst. 91 (2016) 84–92.
[41] W.H. Shu, H. Shen, Incremental feature selection based on rough set in dynamic incomplete data, Pattern Recognit. 47 (12) (2014) 3890–3906.
[42] W.H. Shu, W.B. Qian, An incremental approach to attribute reduction from dynamic incomplete decision systems in rough set theory, Data Knowl. Eng. 100 (2015) 116–132.
[43] A. Skowron, C. Rauszer, The discernibility matrices and functions in information systems, in: R. Slowinski (Ed.), Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht, 1992.

[44] A.H. Tan, J.J. Li, Y.J. Lin, G.P. Lin, Matrix-based set approximations and reductions in covering decision information systems, Int. J. Approximate Reasoning 59 (2015) 68–80.

[45] A.H. Tan, J.J. Li, G.P. Lin, Y.J. Lin, Fast approach to knowledge acquisition in covering information systems using matrix operations, Knowl. Based Syst. 79 (2015) 90–98.

[46] S. Wang, T.R. Li, C. Luo, H. Fujita, Efficient updating rough approximations with multi-dimensional variation of ordered data, Inf. Sci. 372 (2016) 690–708.

[47] F. Wang, J.Y. Liang, C.Y. Dang, Attribute reduction for dynamic data sets, Appl. Soft Comput. 13 (2013) 676–689.

[48] F. Wang, J.Y. Liang, Y.H. Qian, Attribute reduction: a dimension incremental strategy, Knowl. Based Syst. 39 (2013) 95–108.

[49] C.Z. Wang, M.W. Shao, B.Q. Sun, Q.H. Hu, An improved attribute reduction scheme with covering based rough sets, Appl. Soft Comput. 26 (2015) 235–243.

[50] S.P. Wang, W. Zhu, Q.H. Zhu, F. Min, Characteristic matrix of covering and its application to boolean matrix decomposition and axiomatization, Inf. Sci. 263 (1) (2014) 186–197.

[51] W.Z. Wu, Attribute reduction based on evidence theory in incomplete decision systems, Inf. Sci. 178 (2008) 1355–1371.

[52] J.F. Xu, D.Q. Miao, Y.J. Zhang, Z.F. Zhang, A three-way decisions model with probabilistic rough sets for stream computing, Int. J. Approximate Reasoning 88 (2017) 1–22.

[53] W.H. Xu, X.Y. Zhang, J.M. Zhong, Attribute reduction in ordered information systems based on evidence theory, Knowl. Inf. Syst. 25 (2010) 169–184.

[54] B. Yang, B.Q. Hu, On some types of fuzzy covering-based rough sets, Fuzzy Sets Syst. 312 (2017) 36–65.

[55] B. Yang, B.Q. Hu, A fuzzy covering-based rough set model and its generalization over fuzzy lattice, Inf. Sci. 367 (2016) 463–486.

[56] T. Yang, Q.G. Li, Reduction about approximation spaces of covering generalized rough sets, Int. J. Approximate Reasoning 51 (3) (2010) 335–345.

[57] Y.Y. Yang, D.G. Chen, H. Wang, E.C.C. Tsang, D.L. Zhang, Fuzzy rough set based incremental attribute reduction from dynamic data with sample arriving, Fuzzy Sets Syst. 312 (2017) 66–86.

[58] X.B. Yang, Y. Qi, H.L. Yu, X.N. Song, J.Y. Yang, Updating multigranulation rough approximations with increasing of granular structures, Knowl. Based Syst. 64 (2014) 59–69.

[59] X.B. Yang, M. Zhang, H.L. Dou, J.Y. Yang, Neighborhood systems-based rough sets in incomplete information system, Knowl. Based Syst. 24 (6) (2011) 858–867.

[60] Y.Y. Yao, Relational interpretations of neighborhood operators and rough set approximation operators, Inf. Sci. 111 (1) (1998) 239–259.

[61] Y.Y. Yao, B.X. Yao, Covering based rough set approximations, Inf. Sci. 200 (2012) 91–107.

[62] W. Zakowski, Approximations in the space (u, π), Demonstratio Math. 16 (1983) 761–769.

[63] J.B. Zhang, T.R. Li, H.M. Chen, Composite rough sets for dynamic data mining, Inf. Sci. 257 (2014) 81–100.

[64] 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.

[65] B.W. Zhang, F. Min, D. Ciucci, Representative-based classification through covering-based neighborhood rough sets, Appl. Intell. 43 (4) (2015) 840–854.

[66] J.B. Zhang, J.S. Wong, Y. Pan, T.R. Li, A parallel matrix-based method for computing approximations in incomplete information systems, IEEE Trans. Knowl. Data Eng. 27 (2) (2015) 326–339.

[67] X.Y. Zhang, J.Z. Zhou, J. Guo, Q. Zou, Z.W. Huang, Vibrant fault diagnosis for hydroelectric generator units with a new combination of rough sets and support vector machine, Expert Syst. Appl. 39 (2012) 2621–2628.

[68] K. Zheng, J. Hu, Z.F. Zhan, J. Ma, J. Qi, An enhancement for heuristic attribute reduction algorithm in rough set, Expert Syst. Appl. 41 (2014) 6748–6754.

[69] P. Zhu, Covering rough sets based on neighborhoods: an approach without using neighborhoods, Int. J. Approximate Reasoning 52 (3) (2011) 461–472.

[70] W. Zhu, Relationship among basic concepts in covering-based rough sets, Inf. Sci. 179 (14) (2009) 2478–2486.

[71] W. Zhu, Relationship between generalized rough sets based on binary relation and coverings, Inf. Sci. 179 (3) (2009) 210–225.