Knowledge-Based Systems 140 (2018) 142–157

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Discernibility matrix based incremental attribute reduction for dynamic data

Wei Wei[a,*], Xiaoying Wu[a], Jiye Liang[a,*], Junbiao Cui[a], Yijun Sun[b,c]

[a] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan, Shanxi 030006, China

[b] Department of Microbiology and Immunology, State University of New York at Buffalo, Buffalo, NY14201, USA

[c] Department of Computer Science and Engineering, Department of Biostatistics, State University of New York at Buffalo, Buffalo, NY14201, USA

ARTICLE INFO

Article history:
Received 1 February 2017
Revised 27 October 2017
Accepted 31 October 2017
Available online 2 November 2017

Keywords:
Attribute reduction
Discernibility matrix
Incremental algorithm
Dynamic data

ABSTRACT

Dynamic data, in which the values of objects vary over time, are ubiquitous in real applications. Although researchers have developed a few incremental attribute reduction algorithms to process dynamic data, the reducts obtained by these algorithms are usually not optimal. To overcome this deficiency, in this paper, we propose a discernibility matrix based incremental attribute reduction algorithm, through which all reducts, including the optimal reduct, of dynamic data can be incrementally acquired. Moreover, to enhance the efficiency of the discernibility matrix based incremental attribute reduction algorithm, another incremental attribute reduction algorithm is developed based on the discernibility matrix of a compact decision table. Theoretical analyses and experimental results indicate that the latter algorithm requires much less time to find reducts than the former, and that the same reducts can be output by both.

1. Introduction

Attribute reduction, which is considered an important type of rough set theory based feature selection method [5,6,31], aims to select the attributes that retain the discriminatory ability represented by the attribute set of a dataset prior to decision-making [20–23,32]. Researchers have proposed numerous algorithms for implementing attribute reductions [1,5,26,33,34,37,38,40,41,46] based on a discernibility matrix, which is a type of representative method [18,39,47]. Through a discernibility matrix based attribute reduction, all the reducts can be obtained, which is useful to obtain the minimal reduct and generate a subspace for ensemble learning. It should be noted that most of the algorithms mentioned above are only suitable for static datasets.

However, with the rapid development of information technology, three types of datasets, whose object set, attribute set, or attribute values of objects evolve over time, are ubiquitous in many practical applications [3,9,19]. To process these types of datasets, researchers have developed some incremental attribute reduction algorithms over the last two decades. These algorithms are mainly devised based on elementary sets [4], a positive region [28,30,42], information entropy and knowledge granularity [7,14,36], a discernibility matrix [44,45], or a dominance matrix [8]. In addition, Xu et al. [43] presented an incremental algorithm based on 0–1 integer programming. The algorithms above were all developed for use with a complete decision table. Zhang et al. [48] provided a matrix representation of the lower and upper approximations in a set-valued information system. Through an analysis of the variations of the relation matrix resulting from the system variance over time, an incremental approach was introduced to update the rough set approximations, through which the updated reducts can be easily obtained. Liu et al. [15–17] constructed three matrices, based on which some incremental attribute reduction algorithms have been put forward. Chen et al. [2] proposed an equivalent representation of a β-upper (lower) distribution reduct and β-upper (lower) distribution discernibility matrix by means of two Boolean row vectors, and based on these representations, developed a non-incremental algorithm and an incremental algorithm for finding one β-upper (lower) distribution reduct. Nevertheless, most algorithms have aimed at datasets with an object set or attribute set varying over time, and rarely refer to data with attribute values

* Corresponding authors.
E-mail addresses: weiwei@sxu.edu.cn (W. Wei), 1652851969@qq.com (X. Wu), ljy@sxu.edu.cn (J. Liang), 945546899@qq.com (J. Cui), yijunsun@buffalo.edu (Y. Sun).

https://doi.org/10.1016/j.knosys.2017.10.033
0950-7051/

evolving over time. In this paper, we thus focus on attribute reduction for the third type of dataset, i.e., datasets with dynamically varying attribute values, which can be called dynamic datasets [29,35].

To facilitate the following discussion, we review some possible situations associated with dynamic datasets [36]. One situation is that a dataset has some incorrect values, which need to be replaced to obtain the correct output. Another situation is that data we captured gradually increase in amount over time, although the size of dataset we are interested in does not change. We can thus obtain an input dataset for the next moment by slightly modifying an interested dataset at one moment. The other situation is that some useless data should be directly updated using the latest or real-time data because any dated data in a database are often useless in applications such as stock analysis, tests for disease, and annual worker appraisals. In fact, all these types of situations can be considered a change in object attribute values in the dataset.

In [35], Wang et al. introduced a type of incremental attribute reduction algorithm based on entropies for dynamic datasets. With these algorithms, the incremental changing mechanism of three representative entropies [10,11,13,24,25,27], which are usually employed using heuristic attribute reduction, was analyzed. The corresponding incremental reduction algorithms were designed by means of this mechanism. These algorithms actually leverage a similar rationale as the incremental changing mechanism of entropy in [12,14], but the mechanism was modified for dynamic data. In fact, there also exist numerous incomplete dynamic datasets in real-world applications. To efficiently acquire reducts of this type of dataset, in [28–30], Shu and Shen presented three incremental attribute reduction algorithms respectively for three type of datasets varying with time. Among them, the algorithm in [29] aims at dynamic datasets, in which a dynamic changing mechanism of the positive region was proposed to compute a new positive region when the attribute values of an object set vary dynamically. Based on this mechanism, the authors developed two incremental attribute reduction algorithms for incomplete dynamic datasets. Nevertheless, all the incremental algorithms mentioned above are heuristic, and thus, only one reduct of dynamic data, which could contain a few redundant attributes, can be obtained. Moreover, if all reducts of a dynamic dataset can be obtained, we can achieve a number of diverse ensembles, which are beneficial for ensemble learning or group decision-making in certain real-world applications. To acquire all reducts of a dynamic dataset, in this paper, we first developed an incremental attribute reduction algorithm based on a discernibility matrix. Furthermore, inspired by the idea of a compacted decision table, as described in [40], a new compacted decision table and three kinds of discernibility matrices are introduced to design a more efficient incremental algorithm for attribute reduction. The algorithm not only reduces the time consumption of a discernibility matrix based incremental attribute reduction algorithm, it also obtains all reducts of a dynamic dataset.

The remainder of this paper is organized as follows. Section 2 mainly reviews some preliminaries regarding a rough set and discernibility matrix. In Section 3, a discernibility matrix based attribute reduction algorithm is described. In Section 4, a new compacted decision table is defined, three discernibility matrices based on the compacted decision table are introduced, and an attribute reduction algorithm based on the discernibility matrix of a compacted decision table is devised. In Section 5, to demonstrate the effectiveness of the incremental reduction method based on the proposed compacted decision table, we further clarify the relationship between reducts derived from an updated decision table and from its compacted version. In Section 6, extensive experiments carried out to illustrate the efficiency and effectiveness of the proposed algorithms are described. Section 7 provides some concluding remarks.

2. Preliminaries

2.1. Rough set and discernibility matrix

In rough set theory, a basic knowledge expression method, i.e., an information system, is a 4-tuple $S=(U,A,V,f)$ (for short $S=(U,A)$), where U is a non-empty and finite set of objects, called a universe; A is a non-empty and finite set of attributes; V_a is the domain of the attribute a, $V=\bigcup_{a\in A}V_a$; and $f_S: U\times A=V$ is a function, $f_S(x,a)\in V_a$ $(a\in A)$.

For a given information system $S=(U,A,V,f)$, each attribute subset $B\subseteq A$ determines a binary indiscernibility relation: $R_B=\{(x,y)\in U\times U \mid f_S(x,a)=f_S(y,a), \forall a\in B\}$, $f_S(x,a)$ and $f_S(y,a)$ denoting the values of x and y with respect to the attribute a, respectively, and $f_S(x,C)=\cup_{a\in C}\{f_S(x,a)\}$. The relation R_B partitions U into some equivalence classes given by $U/R_B=\{[x]_B|x\in U\}$, where $[x]_B$ is the equivalence class determined by x with respect to B, i.e., $[x]_B=\{y\in U \mid (x,y)\in R_B\}$. Moreover, for any $Y\subseteq U$, $(\overline{B}(Y),\underline{B}(Y))$ is the rough set of Y with respect to B, where $\underline{B}(Y)$ and $\overline{B}(Y)$ are the lower and upper approximations of Y, respectively, and $\underline{B}(Y)=\{x \mid [x]_B\subseteq Y\}$ and $\overline{B}(Y)=\{x \mid [x]_B\cap Y\neq\emptyset\}$.

To describe a classification problem, an information system is modified into a decision table $DT=(U,C\cup\{d\},V,f)$, in which C is called a condition attribute set, and $\{d\}$ is called a decision attribute. To facilitate the development of this study, $V_d=\{v_{d_1},v_{d_2},\ldots,v_{d_l}\}$ was employed to represent the domain of the attribute d. Let $B\subseteq C$, $U/\{d\}=\{Y_1,Y_2,\cdots,Y_n\}$, the lower and upper approximations of the decision attribute $\{d\}$ are defined as $\underline{B}\{d\}=\{\underline{B}Y_1,\underline{B}Y_2,\cdots,\underline{B}Y_n\}$ and $\overline{B}\{d\}=\{\overline{B}Y_1,\overline{B}Y_2,\cdots,\overline{B}Y_n\}$. Let $POS_B(\{d\})=\bigcup_{i=1}^{n}\underline{B}Y_i$, which is called the positive region of $\{d\}$ with respect to B. Moreover, the objects in a positive region make up the consistent part of a decision table, and the other objects comprise the inconsistent part. If $POS_C(\{d\})$ of a decision table is equal to U, it is called a consistent decision table.

In the following, we review three representative discernibility matrices, with regard to a positive region, Shannon entropy, and complement entropy, respectively.

Definition 2.1. [45] Let $DT=(U,C\cup\{d\})$ be a decision table, C be the condition attribute set, and d be the decision attribute. In terms of a positive region, the discernibility matrix is defined as $\mathbf{M}_{DT}^{P}=\{m_{ij}^{P}\}$, where

$$m_{ij}^{P}=\begin{cases}\{c\in C: f(x_i,c)\neq f(x_j,c)\},\ f(x_i,d)\neq f(x_j,d)\text{ and }x_i,x_j\in U_1\\ \{c\in C: f(x_i,c)\neq f(x_j,c)\},\ x_i\in U_1, x_j\in U_2\\ \emptyset,\ otherwise\end{cases},$$

U_1 is the consistent part of the decision table DT and U_2 is the inconsistent part DT.

Its corresponding discernibility function in the sense of a positive region is $\mathcal{F}(\mathbf{M}_{DT}^{P})=\bigwedge\left\{\bigvee(m_{ij}^{P}) \mid \forall x_i,x_j\in U, m_{ij}^{P}\neq\emptyset\right\}$.

In Ref[45]., Yang et al. proposed an incremental attribute reduction when an object was added into a decision table. However, it can only be used to find the set of positive region reducts because their attribute reduction algorithm is based on the discernibility matrix in terms of the positive region. In [39], we developed two new discernibility matrices in terms of Shannon entropy and complement entropy, through which it is easy to extend the algorithm in [45] to compute reducts based on these two entropies. The two matrices are defined as follows.

Table 1
Sixteen changes of the equivalent classes including x_v and x'_v for a decision table.

	$[x_q]_C = \emptyset [x'_q]_C$ is consistent	$[x_q]_C$ is consistent $[x'_q]_C$ is consistent	$[x_q]_C$ is consistent $[x'_q]_C$ is inconsistent	$[x_q]_C$ is inconsistent $[x'_q]_C$ is inconsistent
$[x_p]_C$ is consistent $[x'_p]_C = \emptyset$	T1	T2	T3	T4
$[x_p]_C$ is consistent $[x'_p]_C$ is consistent	T5	T6	T7	T8
$[x_p]_C$ is inconsistent $[x'_p]_C$ is consistent	T9	T10	T11	T12
$[x_p]_C$ is inconsistent $[x'_p]_C$ is inconsistent	T13	T14	T15	T16

Definition 2.2. [39] Let $DT = (U, C \cup \{d\})$ be a decision table, C be the condition attribute set, and be d a decision attribute. The discernibility matrix in terms of Shannon entropy is defined as $\mathbf{M}^S_{DT} = \{m^S_{ij}\}$, where

$$m^S_{ij} = \begin{cases} \{c \in C : f(x_i, c) \neq f(x_j, c)\}, f(x_i, d) \neq f(x_j, d) \text{ and } x_i, x_j \in U_1 \\ \{c \in C : f(x_i, c) \neq f(x_j, c)\}, x_i \in U_1, x_j \in U_2 \\ \{c \in C : f(x_i, c) \neq f(x_j, c)\}, \mu_{ik} \neq \mu_{jk}, \exists Y_k \in U/\{d\}, \text{ and } x_i, x_j \in U_2 \\ \emptyset, \textit{otherwise} \end{cases},$$

$\mu_{ik} = \frac{|[x_i]_C \cap Y_k|}{|[x_i]_C|}$, $\mu_{jk} = \frac{|[x_j]_C \cap Y_k|}{|[x_j]_C|}$, $[x_i]_C \in U/C$ and $[x_j]_C \in U/C$.

Its corresponding discernibility function is $\mathcal{F}(\mathbf{M}^S_{DT}) = \bigwedge \left\{ \bigvee (m^S_{ij}) \mid \forall x_i, x_j \in U, m^S_{ij} \neq \emptyset \right\}$.

Definition 2.3. [39] Let $DT = (U, C \cup \{d\})$ be a decision table, C be the condition attribute set, and d be a decision attribute. The discernibility matrix in the sense of complement entropy is defined as $\mathbf{M}^C_{DT} = \{m^C_{ij}\}$, where

$$m^C_{ij} = \begin{cases} \{c \in C : f(x_i, c) \neq f(x_j, c)\}, f(x_i, d) \neq f(x_j, d) \text{ and } x_i, x_j \in U_1 \\ \{c \in C : f(x_i, c) \neq f(x_j, c)\}, x_i \in U_1, x_j \in U_2 \\ \{c \in C : f(x_i, c) \neq f(x_j, c)\}, x_i, x_j \in U_2 \\ \emptyset, \textit{otherwise} \end{cases}$$

Its corresponding discernibility function is $\mathcal{F}(\mathbf{M}^C_{DT}) = \bigwedge \left\{ \bigvee (m^C_{ij}) \mid \forall x_i, x_j \in U, m^C_{ij} \neq \emptyset \right\}$.

3. Discernibility matrix based incremental attribute reduction algorithm

In this section, to implement a discernibility matrix based incremental attribute reduction for a dynamic dataset, we analyze how the discernibility matrix of a decision table is updated when certain object's values varies over time. The change in discernibility matrix of a decision table may result from a variety of equivalent classes in the decision table. Thus, the change to these equivalent classes will be investigated as follows.

For the development of the analyses, without a loss of generality, we suppose that $DT' = \{U', C \cup \{d\}\}$ is a decision table evolved from $DT = \{U, C \cup \{d\}\}$, where $U = \cup_{i=1}^n \{x_i\}$, $U' = \cup_{i=1}^n \{x'_i\}$, $f_{DT}(x_v, C) \neq f_{DT'}(x'_v, C)$, and $f_{DT}(x_j, C) = f_{DT'}(x'_j, C)$ for $1 \leq j \leq n (j \neq v)$. In addition, we suppose that in DT, $x_v \in [x_p]_C$, and in DT', $[x'_p]_C = [x_p]_C - \{x_v\}$, and if $\exists x'_q$ such that $x'_v \in [x'_q]_C$, then $[x'_q]_C = \{x'_v\} \cup [x_q]_C$; otherwise, $[x'_q]_C = \{x'_v\}$. Thus, sixteen possible changes of the equivalent classes including x_v and x'_v are illustrated in Table 1, and are detailed as follows:

($T1$). $[x_p]_C = \{x_v\}$ is evidently consistent, and after x_v changes to x'_v, $[x'_p]_C = \{x_v\} - \{x_v\} = \emptyset$, $[x'_q]_C = \{x'_v\}$, it is clear that $[x'_q]_C$ is consistent and $[x_q]_C = \emptyset$.

($T2$). $[x_p]_C = \{x_v\}$ is evidently consistent, and after x_v changes to x'_v, $[x'_p]_C = \{x_v\} - \{x_v\} = \emptyset$, $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where both $[x_q]_C$ and $[x'_q]_C$ are clearly consistent.

($T3$). $[x_p]_C = \{x_v\}$ is evidently consistent, and after x_v changes to x'_v, $[x'_p]_C = \{x_v\} - \{x_v\} = \emptyset$, $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where $[x_q]_C$ is consistent, and thus $[x'_q]_C$ is inconsistent.

($T4$). $[x_p]_C = \{x_v\}$ is evidently consistent, and after x_v changes to x'_v, $[x'_p]_C = \{x_v\} - \{x_v\} = \emptyset$, $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where $[x_q]_C$ is inconsistent, and thus $[x'_q]_C$ is also inconsistent.

($T5$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where $[x_p]_C$ is consistent, and thus $[x'_p]_C$ is also consistent; $[x'_q]_C = \{x'_v\}$, and thus $[x'_q]_C$ is consistent and $[x_q]_C = \emptyset$.

($T6$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where both $[x_p]_C$ and $[x'_p]_C$ are consistent; $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where both $[x_q]_C$ and $[x'_q]_C$ are consistent.

($T7$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where $[x_p]_C$ is consistent, and thus $[x'_i]_C$ is consistent; $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where $[x_q]_C$ is consistent and $[x'_q]_C$ is inconsistent.

($T8$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where both $[x_p]_C$ and $[x'_i]_C$ are consistent; $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where $[x_q]_C$ and $[x'_q]_C$ are inconsistent.

($T9$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where $[x_p]_C$ is inconsistent and $[x'_p]_C$ is consistent; $[x'_q]_C = \{x'_v\}$, and thus $[x'_q]_C$ is consistent and $[x_q]_C = \emptyset$.

($T10$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where $[x_p]_C$ is inconsistent and $[x'_p]_C$ is consistent; $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where both $[x_q]_C$ and $[x'_q]_C$ are consistent.

($T11$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where $[x_p]_C$ is inconsistent and $[x'_p]_C$ is consistent; $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where $[x_q]_C$ is consistent and $[x'_q]_C$ is inconsistent.

($T12$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where $[x_p]_C$ is inconsistent and $[x'_p]_C$ is consistent; $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where both $[x_q]_C$ and $[x'_q]_C$ are inconsistent.

($T13$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where both $[x_p]_C$ and $[x'_p]_C$ are inconsistent; $[x'_q]_C = \{x'_v\}$, and thus $[x'_q]_C$ is consistent and $[x_q]_C = \emptyset$.

($T14$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where both $[x_p]_C$ and $[x'_i]_C$ are inconsistent; $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where both $[x_q]_C$ and $[x'_q]_C$ are consistent.

($T15$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where both $[x_p]_C$ and $[x'_i]_C$ are inconsistent. $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where $[x_q]_C$ is consistent and $[x'_q]_C$ is inconsistent.

($T16$). $x_v \in [x_p]_C$, and after x_v changes to x'_v, $[x'_p]_C = [x_p]_C - \{x_v\} \neq \emptyset$, where both $[x_p]_C$ and $[x'_p]_C$ are inconsistent. $[x'_q]_C = \{x'_v\} \bigcup [x_q]_C$, where $[x_q]_C$ is inconsistent, and thus $[x'_q]_C$ is also inconsistent.

Based on the analyses on these sixteen situations mentioned above, a discernibility matrix based incremental attribute reduction algorithm is designed as follows.

Algorithm 1. Discernibility matrix based incremental attribute reduction algorithm for a decision table (DMIAR-DT-Δ)

Input: Decision table $DT = (U, C \cup \{d\})$, the discernibility matrix $\mathbf{M}^{\Delta}_{DT}$ of DT and these objects X_v whose values changed over time, X'_v;

Output: The set of all reducts RED of the decision table.

Step 1: For $x_v \in X_v$, judge which situation the change from x_v to x'_v agrees with.

If the change meets situations ($T1$), ($T2$), ($T5$), or ($T6$), the row of $\mathbf{m}^{\Delta}_{x_v}$ and column of $\mathbf{m}^{x_v}_{\Delta}$ of $\mathbf{M}^{\Delta}_{DT}$ need to be updated.

If the change meets case ($T3$), ($T4$), ($T7$), or ($T8$), modify the rows of $\mathbf{m}^{\Delta}_{k}$, and the columns of $\mathbf{m}^{k}_{\Delta}$ ($x_k \in [x'_j]_C$) of $\mathbf{M}^{\Delta}_{DT}$ need to be updated.

If the change meets cases ($T9$), ($T10$), ($T13$), or ($T14$), then the rows of $\mathbf{m}^{\Delta}_{k}$ and columns of $\mathbf{m}^{k}_{\Delta}$ ($x_k \in [x'_i]_C$) of $\mathbf{M}^{\Delta}_{DT}$, the row of $\mathbf{m}^{\Delta}_{x_v}$, and the column of $\mathbf{m}^{x_v}_{\Delta}$ of $\mathbf{M}^{\Delta}_{DT}$ all need to be updated.

If the change meets cases ($T11$), ($T12$), ($T15$), or ($T16$), then the rows of $\mathbf{m}^{\Delta}_{k}$ and the columns of $\mathbf{m}^{k}_{\Delta}$ ($x_k \in [x'_i]_C$) of $\mathbf{M}^{\Delta}_{DT}$, the rows of $\mathbf{m}^{\Delta}_{l}$, and the columns of $\mathbf{m}^{l}_{\Delta}$ ($x_l \in [x'_j]_C$) of $\mathbf{M}^{\Delta}_{DT}$ all need to be updated.

Step 2: Compute the new discernibility function $\mathcal{F}(\mathbf{M}^{\Delta}_{DT})$;

Step 3: Compute *RED* by means of $\mathcal{F}(\mathbf{M}^{\Delta}_{DT})$;

Step 4: Return *RED* and end.

In this algorithm, $\mathbf{m}^{\Delta}_{x_v}$, $\mathbf{m}^{\Delta}_{l}$ and $\mathbf{m}^{\Delta}_{k}$ are three n-dimension row vectors, $\mathbf{m}^{x_v}_{\Delta}$, $\mathbf{m}^{l}_{\Delta}$ and $\mathbf{m}^{k}_{\Delta}$ are three n-dimension column vectors, $\Delta = \{P, S, C\}$. For convenience, we denote DMIAR-DT-P, DMIAR-DT-S, and DMIAR-DT-C as the different versions of algorithm DMIAR-DT-Δ based on the positive region, Shannon entropy, and complement entropy, respectively.

Time complexity of algorithm DMIAR-DT-Δ is analyzed as follows: When object x_v is changed to x'_v, the number of possible items that vary with the change in a discernibility matrix is $2|U|(|[x_p]_C| + |[x_q]_C|) - (|[x_p]_C|)^2 - (|[x_q]_C|)^2 - 2|[x_p]_C| \times |[x_q]_C|$, and we need a traverse attribute set C to update one item. Thus, the complexity of updating a discernibility matrix is $O\big(|C| \times \big(|U| \times (|[x_p]_C| + |[x_q]_C|) - (|[x_p]_C|)^2 - (|[x_q]_C|)^2 - 2|[x_p]_C| \times |[x_q]_C|\big)\big) = O\big(|C| \times \big(|U|(|[x_p]_C| + |[x_q]_C|)\big)\big)$. When there are $|X_v|$ objects that have been changed, the discernibility matrix will be updated $|X_v|$ times, and it is easy to know that the time complexity is $O\big(|X_v| \times |C| \times \big(2|U| \times (|[x_p]_C| + |[x_q]_C|)\big)\big)$. As is commonly known, the complexity of obtaining all reducts from a discernibility matrix is $O(2^{|C|})$. Therefore, the time complexity of algorithm DMIAR-DT-Δ is $O\big(|X_v| \times |C| \times \big(2|U| \times (|[x_p]_C| + |[x_q]_C|)\big) + 2^{|C|}\big)$.

The space complexity of algorithm DMIAR-DT-Δ is analyzed as follows: The space complexity of storing a decision table is $O(|U| \times |C|)$, the space complexity of storing its discernibility matrix is $O(|U|^2 \times |C|)$, and the space complexity of computing all reducts of the decision table is $O(2^{|C|} \times |C|)$.

4. Discernibility matrices of compacted decision table based incremental attribute reduction

To further enhance the efficiency of the discernibility matrix based incremental attribute reduction algorithm, inspired by the idea of a compacted decision table [40], we introduce a new compacted decision table and its discernibility matrices, and then devise a new incremental attribute reduction method using the compacted decision table.

4.1. A compacted decision table and its discernibility matrices

In [40], we illustrated that there is a large amount of redundant information in a decision table, and introduced a compacted decision table to tackle this problem. a compacted table is very helpful to accelerate static attribute reduction algorithms, but is unsuitable to accomplish attribute reduction for a dynamic dataset. To solve this problem, in this section, we first define a new compacted table. It can preserve all the needed information to obtain reducts of a dynamic dataset, meanwhile eliminating the redundancy resulting from individually computing each object in one equivalent class. Next, in the following section, we introduce three discernibility matrices based on the compacted decision table, which provides an important basis for incremental attribute reduction algorithms.

Table 2
A decision table.

	a_1	a_2	a_3	a_4	d
x_1	1	1	1	1	0
x_2	2	2	2	1	1
x_3	1	1	1	1	0
x_4	1	3	1	3	0
x_5	2	2	2	1	1
x_6	3	1	2	1	0
x_7	2	2	3	2	2
x_8	2	3	1	2	3
x_9	3	1	2	1	1
x_{10}	2	2	3	2	2
x_{11}	3	1	2	1	1
x_{12}	2	3	1	2	3
x_{13}	4	3	4	2	1
x_{14}	2	2	3	2	2
x_{15}	4	3	4	2	2

Definition 4.1. Given a decision table $DT = (U, C \cup \{d\})$, $U = \{x_1, x_2, \cdots, x_n\}$, $U/C = \{X_1, X_2, \cdots, X_m\}$, $V_d = \{v_{d_1}, v_{d_2}, \cdots, v_{d_l}\}$, and a compacted decision table is then defined as $CDT = (CU, C \cup CD)$, where $CU = \{cx_1, cx_2, \cdots, cx_m\}$, $f_{CDT}(cx_i, C) = f_{DT}(X_i, C)$ (i.e. $f_{CDT}(cx_i, c) = f_{DT}(X_i, c)$, for $\forall c \in C$), $CD = \{cd_1, cd_2, \cdots, cd_l\}$, $f_{CDT}(cx_i, cd_j) = \{x \mid f_{DT}(x, d) = v_{d_j}, x \in X_i\}$.

For a given compacted decision table $CDT = (CU, C \cup CD)$, $CU_1 = \{cx_i | \sigma_{CD}(cx_i) = 1, cx_i \in CU\}$ indicates its consistent part, and $CU_2 = CU - CU_1$ indicates its inconsistent part, where $\sigma_{CD}(cx_i) = \bigcup_{k=1}^{l} \{cd_k | f_{CDT}(cx_i, cd_k) \neq \emptyset\}$. To illustrate what a concrete compacted decision table is, we employ the following example.

Example 4.1. $DT = (U, C \cup D)$ is a decision table (shown in Table 1), $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}\}$, $U/C = \{\{x_1, x_3\}, \{x_2, x_5\}, \{x_4\}, \{x_6, x_9, x_{11}\}, \{x_7, x_{10}, x_{14}\}, \{x_8, x_{12}\}, \{x_{13}, x_{15}\}\}$, $V_d = \{0, 1, 2, 3\}$. Then, based on the definition of the compacted decision table, we obtain $CU = \{cx_1, cx_2, cx_3, cx_4, cx_5, cx_6, cx_7\}$, $f_{CDT}(cx_1, C) = f_{DT}(x_1, C)$, $f_{CDT}(cx_2, C) = f_{DT}(x_2, C)$, $f_{CDT}(cx_3, C) = f_{DT}(x_4, C)$, $f_{CDT}(cx_4, C) = f_{DT}(x_6, C)$, $f_{CDT}(cx_5, C) = f_{DT}(x_7, C)$, $f_{CDT}(cx_6, C) = f_{DT}(x_8, C)$, $f_{CDT}(cx_7, C) = f_{DT}(x_{13}, C)$. Moreover, it is easy to see that $f_{CDT}(cx_1, cd_0) = \{x_1, x_3\}$, $f_{CDT}(cx_1, cd_1) = f_{CDT}(cx_1, cd_2) = f_{CDT}(cx_1, cd_3) = \emptyset$, $f_{CDT}(cx_2, cd_1) = \{x_2, x_5\}$, $f_{CDT}(cx_2, cd_0) = f_{CDT}(cx_2, cd_2) = f_{CDT}(cx_2, cd_3) = \emptyset$, $f_{CDT}(cx_3, cd_0) = \{x_4\}$, $f_{CDT}(cx_3, cd_1) = f_{CDT}(cx_3, cd_2) = f_{CDT}(cx_3, cd_3) = \emptyset$, $f_{CDT}(cx_4, cd_0) = \{x_6\}$, $f_{CDT}(cx_4, cd_1) = \{x_9, x_{11}\}$, $f_{CDT}(cx_4, cd_2) = f_{CDT}(cx_4, cd_3) = \emptyset$, $f_{CDT}(cx_5, cd_2) = \{x_7, x_{10}, x_{14}\}$, $f_{CDT}(cx_5, cd_0) = f_{CDT}(cx_5, cd_1) = f_{CDT}(cx_5, cd_3) = \emptyset$, $f_{CDT}(cx_6, cd_3) = \{x_8, x_{12}\}$, $f_{CDT}(cx_6, cd_0) = f_{CDT}(cx_6, cd_1) = f_{CDT}(cx_6, cd_2) = \emptyset$, $f_{CDT}(cx_7, cd_1) = \{x_{13}\}$, $f_{CDT}(cx_7, cd_2) = \{x_{15}\}$, $f_{CDT}(cx_7, cd_0) = f_{CDT}(cx_7, cd_3) = \emptyset$. Table 2 presents the compacted version of Table 1.

Based on Definition 4.1, three new discernibility matrices that can capture all the discernibility information of a compacted decision table with regard to the positive region, Shannon entropy, and complement entropy are proposed as follows:

Table 3
A decision table compacted from Table 1.

	a_1	a_2	a_3	a_4	cd_0	cd_1	cd_2	cd_3
cx_1	1	1	1	1	$\{x_1, x_3\}$	$\emptyset$	$\emptyset$	$\emptyset$
cx_2	2	2	2	1	$\emptyset$	$\{x_2, x_5\}$	$\emptyset$	$\emptyset$
cx_3	1	3	1	3	$\{x_4\}$	$\emptyset$	$\emptyset$	$\emptyset$
cx_4	3	1	2	1	$\{x_6\}$	$\{x_9, x_{11}\}$	$\emptyset$	$\emptyset$
cx_5	2	2	3	2	$\emptyset$	$\emptyset$	$\{x_7, x_{10}, x_{14}\}$	$\emptyset$
cx_6	2	3	1	2	$\emptyset$	$\emptyset$	$\emptyset$	$\{x_8, x_{12}\}$
cx_7	4	3	4	2	$\emptyset$	$\{x_{13}\}$	$\{x_{15}\}$	$\emptyset$

Definition 4.2. Given a decision table $DT = (U, C \cup \{d\})$ and its compacted version $CDT = (CU, C \cup CD)$, a discernibility matrix of CDT in terms of the positive region is defined as $\mathbf{M}_{CDT}^{P} = \{cm_{pq}^{P}\}$, where

$$cm_{pq}^{P} = \begin{cases} \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & \{cd_k | f_{CDT}(cx_p, cd_k) \neq \emptyset\} \neq \{cd_k | f_{CDT}(cx_q, cd_k) \neq \emptyset\} \text{ and } cx_p, cx_q \in CU_1 \\ \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & cx_p \in CU_1, cx_q \in CU_2 \\ \emptyset, & otherwise \end{cases}.$$

Example 4.2. According to Definition 4.2, the discernibility matrix in Table 3, with regard to the positive region, is given as follows:

$$\mathbf{M}_{CDT}^{P} = \begin{pmatrix} \emptyset & \{a_1,a_2,a_3\} & \emptyset & \{a_1,a_3\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \{a_1,a_2,a_3\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2\} & \{a_3,a_4\} & \{a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \emptyset & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_4\} & \{a_1,a_3,a_4\} \\ \{a_1,a_3\} & \{a_1,a_2\} & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \emptyset \\ \{a_1,a_2,a_3,a_4\} & \{a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_2,a_3\} & \{a_1,a_2,a_3\} \\ \{a_1,a_2,a_4\} & \{a_2,a_3,a_4\} & \{a_1,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_2,a_3\} & \emptyset & \{a_1,a_3\} \\ \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_3,a_4\} & \emptyset & \{a_1,a_2,a_3\} & \{a_1,a_3\} & \emptyset \end{pmatrix}.$$

Definition 4.3. Given a decision table $DT = (U, C \cup \{d\})$ and its compacted version $CDT = (CU, C \cup CD)$, a discernibility matrix in terms of the Shannon entropy is defined as $\mathbf{M}_{CDT}^{S} = \{cm_{pq}^{S}\}$, where

$$cm_{pq}^{S} = \begin{cases} \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & \{cd_k | f_{CDT}(cx_p, cd_k) \neq \emptyset\} \neq \{cd_k | f_{CDT}(cx_q, cd_k) \neq \emptyset\} \text{ and } cx_p, cx_q \in CU_1 \\ \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & cx_p \in CU_1, cx_q \in CU_2 \\ \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & \dfrac{|f_{CDT}(cx_p, cd_k)|}{\bigcup_{i=1}^{l} |f_{CDT}(cx_p, cd_i)|} \neq \dfrac{|f_{CDT}(cx_q, cd_k)|}{\bigcup_{j=1}^{l} |f_{CDT}(cx_q, cd_j)|}, \exists cd_k \in CD, \text{ and } cx_p, cx_q \in CU_2 \\ \emptyset, & otherwise \end{cases}.$$

Example 4.3. According to Definition 4.3, the discernibility matrix of Table 3 with regard to the Shannon entropy is given as follows:

$$\mathbf{M}_{CDT}^{S} = \begin{pmatrix} \emptyset & \{a_1,a_2,a_3\} & \emptyset & \{a_1,a_3\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \{a_1,a_2,a_3\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2\} & \{a_3,a_4\} & \{a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \emptyset & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_4\} & \{a_1,a_3,a_4\} \\ \{a_1,a_3\} & \{a_1,a_2\} & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \{a_1,a_2,a_3,a_4\} & \{a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_2,a_3\} & \{a_1,a_2,a_3\} \\ \{a_1,a_2,a_4\} & \{a_2,a_3,a_4\} & \{a_1,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_2,a_3\} & \emptyset & \{a_1,a_3\} \\ \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3\} & \{a_1,a_3\} & \emptyset \end{pmatrix}.$$

Definition 4.4. Given a decision table $DT = (U, C \cup \{d\})$ and its compacted version $CDT = (CU, C \cup CD)$, a discernibility matrix in terms of the complement entropy is defined as $\mathbf{M}_{CDT}^{C} = \{cm_{pq}^{C}\}$, where

$$cm_{pq}^{C} = \begin{cases} \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & \{cd_k | f_{CDT}(cx_p, cd_k) \neq \emptyset\} \neq \{cd_k | f_{CDT}(cx_q, cd_k) \neq \emptyset\} \text{ and } cx_p, cx_q \in CU_1 \\ \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & cx_p \in CU_1, cx_q \in CU_2 \\ \{c \in C : f_{CDT}(cx_p, c) \neq f_{CDT}(cx_q, c)\}, & cx_p, cx_q \in CU_2 \\ \emptyset, & otherwise \end{cases}.$$

Example 4.4. According to Definition 4.4, the discernibility matrix of Table 3 with regard to the complement entropy is given as follows:

$$\mathbf{M}_{CDT}^{C} = \begin{pmatrix} \emptyset & \{a_1,a_2,a_3\} & \emptyset & \{a_1,a_3\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \{a_1,a_2,a_3\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2\} & \{a_3,a_4\} & \{a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \emptyset & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_4\} & \{a_1,a_3,a_4\} \\ \{a_1,a_3\} & \{a_1,a_2\} & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} \\ \{a_1,a_2,a_3,a_4\} & \{a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \emptyset & \{a_2,a_3\} & \{a_1,a_2,a_3\} \\ \{a_1,a_2,a_4\} & \{a_2,a_3,a_4\} & \{a_1,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_2,a_3\} & \emptyset & \{a_1,a_3\} \\ \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_3,a_4\} & \{a_1,a_2,a_3,a_4\} & \{a_1,a_2,a_3\} & \{a_1,a_3\} & \emptyset \end{pmatrix}.$$

Table 4
Sixteen possible changes of objects that are compacted from the equivalent classes including x_v and x'_v.

	$obj(cx_q)=0, \lvert\sigma_{CDT'}(cx'_q)\rvert=1$	$\lvert\sigma_{CDT}(cx_q)\rvert=1, \lvert\sigma_{CDT'}(cx'_q)\rvert=1$	$\lvert\sigma_{CDT}(cx_q)\rvert=1, \lvert\sigma_{CDT'}(cx'_q)\rvert>1$	$\lvert\sigma_{CDT}(cx_q)\rvert>1, \lvert\sigma_{CDT'}(cx'_q)\rvert>1$
$\lvert\sigma_{CDT}(cx_p)\rvert=1, obj(cx'_p)=0$	CT1	CT2	CT3	CT4
$\lvert\sigma_{CDT}(cx_p)\rvert=1, \lvert\sigma_{CDT'}(cx'_p)\rvert=1$	CT5	CT6	CT7	CT8
$\lvert\sigma_{CDT}(cx_p)\rvert>1, \lvert\sigma_{CDT'}(cx'_p)\rvert=1$	CT9	CT10	CT11	CT12
$\lvert\sigma_{CDT}(cx_p)\rvert>1, \lvert\sigma_{CDT'}(cx'_p)\rvert>1$	CT13	CT14	CT15	CT16

4.2. Incremental attribute reduction algorithm

An incremental attribute reduction algorithm should be developed based on the difference between before and after a compacted decision table variation. Thus, it is inevitable to answer the question regarding how a compacted decision table changes after a variation of the attribute values appears in its original version. To reach this end, we first analyze the possible changes of a compacted decision table caused by a change in attribute values in its original version.

For the development described in this section, without a loss of any generality, we suppose that $DT'=(U', C\cup\{d\})$ is a decision table evolved from $DT=(U, C\cup\{d\})$, where $U=\cup_{i=1}^{n}\{x_i\}$, $U'=\cup_{i=1}^{n}\{x'_i\}$, $f_{DT}(x_v, C)\neq f_{DT'}(x'_v, C)$, and $f_{DT}(x_j, C)=f_{DT'}(x'_j, C)$ for $1\leq j\leq n(j\neq v)$. We then suppose that $CDT=(CU, C\cup CD)$ is a decision table compacted from $DT=(U, C\cup\{d\})$, and $CDT'=(CU', C\cup CD)$ is updated from CDT owing to the change of $x_v\in U$ into $x'_v\in U'$. Furthermore, we suppose that in CDT, there exists $cx_p\in CU$ such that $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$, and in CDT', $f_{CDT'}(x'_v, C)\neq f_{CDT'}(cx'_p, C)$ (cx'_p evolved from cx_p), and if $\exists cx'_q$ such that $f_{DT'}(x'_v, C)=f_{DT'}(cx'_q, C)$, we suppose $f_{CDT'}(cx'_q, C)=f_{CDT'}(x'_v, C)$ and $|obj(cx'_q)|>1$; otherwise, we suppose $f_{CDT'}(cx'_q, C)=f_{CT'}(x'_v, C)$ and $|obj(cx'_q)|=1$, where, $obj(cx'_i)=\{x|x\in U', f_{CDT'}(cx'_i, C)=f_{DT'}(x, C)\}$.

In a decision table, the change in attribute values of an object x_v may result in changes to its compacted version. Based on the status of the equivalent classes related with x_v before and after the change of a compacted decision table, sixteen possible changes (shown in Table 4) are described in detail as follows:

($CT1$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $obj(cx_p)=1$, we thus have $|\sigma_{CDT}(cx_p)|=1$; in addition, after x_v changes into x'_v, $obj(cx'_p)=0$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, and $obj(cx_q)=0$.

($CT2$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $obj(cx_p)=1$, we thus have $|\sigma_{CDT}(cx_p)|=1$; in addition, after x_v changes into x'_v, $obj(cx'_p)=0$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, and $|\sigma_{CDT}(cx_q)|=1$.

($CT3$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $obj(cx_p)=1$, we thus have $|\sigma_{CDT}(cx_p)|=1$; in addition, after x_v changes into x'_v, $obj(cx'_p)=0$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|>1$, and $|\sigma_{CDT}(cx_q)|=1$.

($CT4$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $obj(cx_p)=1$, we thus have $|\sigma_{CDT}(cx_p)|=1$; in addition, after x_v changes into x'_v, $obj(cx'_p)=0$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|>1$, and $|\sigma_{CDT}(cx_q)|>1$.

($CT5$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|=1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, and $obj(cx_q)=0$.

($CT6$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|=1$, and after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|=1$.

($CT7$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|=1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|>1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|=1$.

($CT8$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|=1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|>1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|>1$.

($CT9$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, and $obj(cx_q)=0$.

($CT10$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|=1$.

($CT11$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|>1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|=1$.

($CT12$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|>1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|>1$.

($CT13$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|>1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, and $obj(cx_q)=0$.

($CT14$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|>1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|=1$.

($CT15$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|=1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|=1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$, and $|\sigma_{CDT}(cx_q)|>1$.

($CT16$). For $x_v\in U$, $f_{DT}(x_v, C)=f_{CDT}(cx_p, C)$ and $|\sigma_{CDT}(cx_p)|>1$; in addition, after x_v changes into x'_v, $f_{DT}(x'_v, C)\neq f_{CDT}(cx'_p, C)$, $|\sigma_{CDT}(cx'_p)|>1$, $f_{DT'}(x'_v, C)=f_{CDT'}(cx'_q, C)$, $|\sigma_{CDT'}(cx'_q)|>1$, $f_{DT}(x_v, C)=f_{CDT}(cx_q, C)$ and $|\sigma_{CDT}(cx_q)|>1$.

Based on these changing situations of a compacted decision table, we devise another discernibility matrix based attribute reduction algorithm as follows.

Algorithm 2. Discernibility matrix based incremental attribute reduction for a compacted decision table (DMIAR-CDT-Δ)

Input: A compacted decision table $CDT=(CU, C\cup CD)$, its discernibility matrix $\mathbf{M}^{\Delta}_{CDT}$, and those objects X_v whose values change over time, X'_v.

Output: All reducts RED of CDT'.

Step 1: For $\forall x_v\in X_v$, search cx_p whose $obj(cx_p)$ includes x_v, and compute cx'_a whose $obj(cx'_a)$ includes x'_v by means of cx_p and x'_v, and search cx_q, which evolves into cx'_a. Then, judge which situation is consistent with the changes in cx_p and cx_q.

If the change agrees with situation ($CT1$), then the row $\mathbf{cm}^{\Delta}_{cx_p}$ and column $\mathbf{cm}^{cx_p}_{\Delta}$ of $\mathbf{M}^{\Delta}_{CDT}$ need to be deleted, and the row $\mathbf{cm}^{\Delta}_{cx'_a}$ and column $\mathbf{cm}^{cx'_a}_{\Delta}$ need to be added into $\mathbf{M}^{\Delta}_{CDT}$.

If the change agrees with the situation ($CT2$), then the row of $\mathbf{cm}^{\Delta}_{cx_p}$ and the column $\mathbf{cm}^{cx_p}_{\Delta}$ of $\mathbf{M}^{\Delta}_{CDT}$ need to be deleted.

If the change agrees with situations ($CT3$) or ($CT4$), then the row $\mathbf{cm}^{\Delta}_{cx_p}$ and column of $\mathbf{cm}^{cx_p}_{\Delta}$ of $\mathbf{M}^{\Delta}_{CDT}$ need to be deleted, and the row $\mathbf{cm}^{\Delta}_{cx_q}$ and column of $\mathbf{cm}^{cx_q}_{\Delta}$ will be updated.

If the change agrees with situation ($CT5$), then the row $\mathbf{cm}^{\Delta}_{cx'_a}$ and column $\mathbf{cm}^{cx'_a}_{\Delta}$ need to be added into $\mathbf{M}^{\Delta}_{CDT}$.

If the change agrees with situation ($CT6$), then the discernibility matrix $\mathbf{M}^{\Delta}_{CDT}$ remains unchanged.

If the change agrees with situations ($CT7$) or ($CT8$), then the row $\mathbf{cm}^{\Delta}_{cx_q}$ and column $\mathbf{cm}^{cx_q}_{\Delta}$ of $\mathbf{M}^{\Delta}_{CDT}$ need to be updated.

If the change agrees with situations ($CT9$) or ($CT13$), then the row of $\mathbf{cm}^{\Delta}_{cx'_a}$ and column of $\mathbf{cm}^{cx'_a}_{\Delta}$ need to be added into $\mathbf{M}^{\Delta}_{CDT}$, and the row of $\mathbf{cm}^{\Delta}_{cx_p}$ and the column of $\mathbf{cm}^{cx_p}_{\Delta}$ need to be updated.

If the change agrees with situations ($CT10$) or ($CT14$), then the row of $\mathbf{cm}^{\Delta}_{cx_p}$ and column of $\mathbf{cm}^{cx_p}_{\Delta}$ need be updated.

If the change agrees with situations ($CT11$), ($CT12$), ($CT15$), or ($CT16$), then the row of $\mathbf{cm}^{\Delta}_{cx_p}$, the column of $\mathbf{cm}^{cx_p}_{\Delta}$, the row of $\mathbf{cm}^{\Delta}_{cx_q}$, and the column of $\mathbf{cm}^{cx_q}_{\Delta}$ of $\mathbf{M}^{\Delta}_{CDT}$ all need to be updated.

Step 2: Compute the new discernibility function $\mathcal{F}(\mathbf{M}^{\Delta}_{CDT})$.

Step 3: Compute *RED* using updated discernibility matrix $\mathcal{F}(\mathbf{M}^{\Delta}_{CDT})$.

Step 4: Return *RED* and end.

where $\mathbf{cm}^{\Delta}_{cx_p}$, $\mathbf{cm}^{\Delta}_{cx'_a}$ and $\mathbf{cm}^{\Delta}_{cx_q}$ are row vectors, and $\mathbf{cm}^{cx_p}_{\Delta}$, $\mathbf{cm}^{cx'_a}_{\Delta}$ and $\mathbf{cm}^{cx_q}_{\Delta}$ are column vectors. In addition, in the algorithm, the parameter Δ equals {*P, S, C*}, i.e., DMIAR-CDT-P, DMIAR-CDT-S, and DMIAR-CDT-C indicate the specific versions of the positive region, Shannon entropy, and complement entropy, respectively.

The time complexity of algorithm DMIAR-DT-Δ is analyzed as follows: Because the equivalent classes $[x_p]_C$ and $[x_q]_C$ related to a change from x_v and x'_v will be compacted into two objects x_p and x_q in *CDT*, respectively, the number of possible items affected by the change in discernibility matrix is $2|CU| \times 2 - 1 - 1 - 2$, and we need traverse attribute set C to update one item. Thus, the complexity of updating a discernibility matrix is $O(|C| \times |CU|)$. In addition, when $|X_v|$ objects are changed, the discernibility matrix will be updated $|X_v|$ times, and thus the time complexity is $O(|X_v| \times |C| \times |CU|)$. The complexity in obtaining all reducts by using a discernibility matrix is $O(2^{|C|})$. Therefore, the time complexity of algorithm DMIAR-DT-Δ is $O\big(|X_v| \times |C| \times |CU| + 2^{|C|}\big)$.

The space complexity of algorithm DMIAR-DT-Δ is analyzed as follows: The space complexity of storing a compacted decision table is $O(|CU| \times |C|)$, the space complexity of storing its discernibility matrix is $O(|CU|^2 \times |C|)$, and the space complexity of computing all reducts of a compacted decision table is $O(2^{|C|} \times |C|)$.

5. Relationship between reducts of a decision table with changing object values and its compacted version

After we obtain reducts from a compacted decision with the value of changing objects over time, it is natural to wonder whether the same reducts can be acquired by the compacted table as compared to the original version. To answer this question, we investigate the relationship between the discernibility function of a decision table and its compacted version, and analyze how a compacted decision table changes with the variation of object values. On basis of these analyses, we finally reveal the relationship between reducts of a decision table with object values varying over time and its compacted version.

5.1. Relationship between the discernibility functions M(DT) and M(CDT)

Because the discernibility matrix of a compacted decision table is defined based on the discernibility matrix of a decision table, we may thus speculate that there should be a certain relationship between these two discernibility matrices, and a relationship between their corresponding discernibility functions. The following theorems are employed to indicate these relationships.

Theorem 5.1. *Given a decision table* $DT = (U, C \cup \{d\})$ *and its compacted version* $CDT = (CU, C \cup CD)$*. The relationship between discernibility functions generated based on DT and CDT is* $\mathcal{F}(\mathbf{M}^P_{DT}) = \mathcal{F}(\mathbf{M}^P_{CDT})$.

Proof. Suppose that $U = \{x_1, x_2, \cdots, x_n\}$ and $CU = \{cx_1, cx_2, \cdots, cx_m\}$. From the definition of a compacted decision table, without a loss of generality, we further suppose that $U/C = \{X_1, X_2, \cdots, X_m\}$, and $f_{DT}(x_{p_i}, C) = f_{CDT}(cx_p, C)$ for $\forall x_{p_i} \in X_p$.

(1) $cx_p,\ cx_q \in CU$, $\{cd_k \in CD | f_{CDT}(cx_p,\ cd_k) \neq \emptyset\} \neq \{cd_k \in CD | f_{CDT}(cx_q, cd_k) \neq \emptyset\}$ and $cx_p,\ cx_q \in CU_1$
In this case, it is easy to obtain that $cx_p, cx_q \in CU_1 \Leftrightarrow x_{p_i}, x_{q_j} \in U_1$, $(x_{p_i} \in X_p, x_{q_j} \in X_q)$, and $\{cd_k \in CD | f_{CDT}(cx_p, cd_k) \neq \emptyset\} \neq \{cd_k \in CD | f_{CDT}(cx_q, cd_k) \neq \emptyset\} \Leftrightarrow f_{DT}(x_{p_i}, d) \neq f_{DT}(x_{q_j}, d)$. We thus have $m^P_{p_i q_j} = cm^P_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(2) $cx_p \in CU_1$, $cx_q \in CU_2$
In this case, we have $cx_p \in CU_1 \Leftrightarrow x_i \in U_1$ for $\forall x_i \in X_p$, and $cx_q \in CU_2 \Leftrightarrow x_q \in U_2$ for $\forall x_j \in X_q$. We thus have $m^P_{p_i q_j} = cm^P_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(3) Otherwise
In this case, it is easy to see that $m^P_{p_i q_j} = cm^P_{pq} = \emptyset$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.
Because $\bigvee(m^P_{p_i q_j}) = cm^P_{pq}$, we have

$$
\begin{aligned}
\mathcal{F}(\mathbf{M}^P_{DT}) &= \bigwedge \left\{ \bigvee(m^P_{p_i q_j}) \mid \forall x_{p_i}, x_{q_j} \in U, m^P_{p_i q_j} \neq \emptyset \right\} \\
&= \bigwedge \left\{ \bigvee(cm^P_{pq}) \mid \forall cx_p, cx_q \in CU, cm^P_{pq} \neq \emptyset \right\} \\
&= \mathcal{F}(\mathbf{M}^P_{CDT}).
\end{aligned}
$$

□

Theorem 5.1 states the discernibility function of a compacted decision table is the same as that of its original version, and thus, all reducts acquired from a decision table that are the same as those acquired from its compacted version can be obtained. Next, the relationship between the discernibility function of a decision table and its compacted version in terms of the Shannon entropy is investigated in the following theorem.

Theorem 5.2. *Given a decision table* $DT = (U, C \cup \{d\})$ *and its compacted version* $CDT = (CU, C \cup CD)$*, the relationship between discernibility functions generated from DT and CDT is* $\mathcal{F}(\mathbf{M}^S_{DT}) = \mathcal{F}(\mathbf{M}^S_{CDT})$.

Proof. Suppose that $U = \{x_1, x_2, \cdots, x_n\}$ and $CU = \{cx_1, cx_2, \cdots, cx_m\}$. From the definition of a compacted decision table, without a loss of generality, we further suppose that $U/C = \{X_1, X_2, \cdots, X_m\}$, and $f_{DT}(x_{p_i}, C) = f_{CDT}(cx_p, C)$ for $\forall x_{p_i} \in X_p$.

(1) $cx_p,\ cx_q \in CU$, $\{cd_k \in CD | f_{CDT}(cx_p,\ cd_k) \neq \emptyset\} \neq \{cd_k \in CD | f_{CDT}(cx_q,\ cd_k) \neq \emptyset\}$ and $cx_p,\ cx_q \in CU_1$.
In this case, it is easy to obtain $cx_p, cx_q \in CU_1 \Leftrightarrow x_{p_i}, x_{q_j} \in U_1, (x_{p_i} \in X_p, x_{q_j} \in X_q)$, and $\{cd_k \in CD | f_{CDT}(cx_p, cd_k) \neq \emptyset\} \neq \{cd_k \in CD | f_{CDT}(cx_q, cd_k) \neq \emptyset\} \Leftrightarrow f_{DT}(x_{p_i}, d) \neq f_{DT}(x_{q_j}, d)$. We thus have $m^S_{p_iq_j} = cm^S_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(2) $cx_p \in CU_1,\ cx_q \in CU_2$
In this case, it is easy to obtain $cx_p \in CU_1 \Leftrightarrow x_p \in U_1$ for $\forall x_i \in X_p$, and $cx_q \in CU_2 \Leftrightarrow x_q \in U_2$ for $\forall x_j \in X_q$. We thus have $m^S_{p_iq_j} = cm^S_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(3) $\exists cd_k \in CD$ such that $\frac{|f_{CDT}(cx_p, cd_k)|}{\bigcup_{i=1}^{l} |f_{CDT}(cx_p, cd_i)|} \neq \frac{|f_{CDT}(cx_q, cd_k)|}{\bigcup_{j=1}^{l} |f_{CDT}(cx_q, cd_j)|}$, and $cx_p,\ cx_q \in CU_2$
In this case, it is easy to obtain $cx_p,\ cx_q \in CU_2 \Leftrightarrow x_p,\ x_q \in U_2$. From the definition of a compacted decision table, we have $f_{CDT}(cx_p, cd_k) = X_p \cap Y_k$ and $f_{CDT}(cx_q, cd_k) = X_q \cap Y_k$, and thus $\exists cd_k \in CD$ such that $\frac{|f_{CDT}(cx_p, cd_k)|}{\bigcup_{i=1}^{l} |f_{CDT}(cx_p, cd_i)|} \neq \frac{|f_{CDT}(cx_q, cd_k)|}{\bigcup_{j=1}^{l} |f_{CDT}(cx_q, cd_j)|}$ $\Leftrightarrow \mu_{pk} = \frac{|X_p \cap Y_k|}{|X_p|} \neq \frac{|X_q \cap Y_k|}{|X_q|} = \mu_{qk}$, $\exists Y_k \in U/\{d\}$. We thus have $m^S_{p_iq_j} = cm^S_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(4) Otherwise
In this case, it is easy to see that $m^S_{p_iq_j} = cm^S_{pq} = \emptyset$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.
Furthermore, because of $\bigvee(m^S_{p_iq_j}) = cm^S_{pq}$, we have

$$\begin{aligned}\mathcal{F}(\mathbf{M}^S_{DT}) &= \bigwedge \left\{ \bigvee (m^S_{p_iq_j}) \mid \forall p_i, q_j \in U, m^S_{p_iq_j} \neq \emptyset \right\} \\ &= \bigwedge \left\{ \bigvee (cm^S_{pq}) \mid \forall cx_p, cx_q \in CU, cm^S_{pq} \neq \emptyset \right\} \\ &= \mathcal{F}(\mathbf{M}^S_{CDT}).\end{aligned}$$

□

From Theorem 5.2, we can see that the discernibility function of a compacted decision table is the same as that of its original version. It is apparent that the reducts derived from a compacted decision table are identical to those from its original version.

Finally, we analyze the relationship between the discernibility function of a decision table and its compacted version in terms of complement entropy.

Theorem 5.3. *Given a decision table $DT = (U, C \cup \{d\})$ and its compacted version $CDT = (CU, C \cup CD)$, the relationship between discernibility matrices generated from DT and CDT is $\mathcal{F}(\mathbf{M}^C_{DT}) = \mathcal{F}(\mathbf{M}^C_{CDT})$.*

Proof. Suppose that $U = \{x_1, x_2, \cdots, x_n\}$ and $CU = \{cx_1, cx_2, \cdots, cx_m\}$. From the definition of a compacted decision table, without a loss of generality, we further suppose that $U/C = \{X_1, X_2, \cdots, X_m\}$, and $f_{DT}(x_{p_i}, C) = f_{CDT}(cx_p, C)$ for $\forall x_{p_i} \in X_p$.

(1) $cx_p,\ cx_q \in CU$, $\{cd_k \in CD | f_{CDT}(cx_p,\ cd_k) \neq \emptyset\} \neq \{cd_k \in CD | f_{CDT}(cx_q,\ cd_k) \neq \emptyset\}$ and $cx_p,\ cx_q \in CU_1$.
In this case, it is easy to obtain $cx_p, cx_q \in CU_1 \Leftrightarrow x_{p_i}, x_{q_j} \in U_1, (x_{p_i} \in X_p, x_{q_j} \in X_q)$, and $\{cd_k \in CD | f_{CDT}(cx_p, cd_k) \neq \emptyset\} \neq \{cd_k \in CD | f_{CDT}(cx_q, cd_k) \neq \emptyset\} \Leftrightarrow f_{DT}(x_{p_i}, d) \neq f_{DT}(x_{q_j}, d)$. We thus have $m^C_{p_iq_j} = cm^C_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(2) $cx_p \in CU_1,\ cx_q \in CU_2$
In this case, it is easy to obtain $cx_p \in CU_1 \Leftrightarrow x_i \in U_1$ for $\forall x_i \in X_p$, and $cx_q \in CU_2 \Leftrightarrow x_q \in U_2$ for $\forall x_j \in X_q$. We thus have $m^C_{p_iq_j} = cm^C_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(3) $cx_p,\ cx_q \in CU_2$
In this case, it is easy to obtain $cx_p,\ cx_q \in CU_2 \Leftrightarrow x_p,\ x_q \in U_2$ for $\forall x_i \in X_p$. We thus have $m^C_{p_iq_j} = cm^C_{pq}$ for $\forall x_{p_i} \in X_p,\ \forall x_{q_j} \in X_q$.

(4) Otherwise
In this case, it is easy to see that $m^C_{p_iq_j} = cm^C_{pq} = \emptyset$,for $\forall x_{p_i} \in X_p$, $\forall x_{q_j} \in X_q$.
Furthermore, because of $\bigvee(m^C_{p_iq_j}) = cm^C_{pq}$, we have

$$\begin{aligned}\mathcal{F}(\mathbf{M}^C_{DT}) &= \bigwedge \left\{ \bigvee (m^C_{p_iq_j}) \mid \forall p_i, q_j \in U, m^C_{p_iq_j} \neq \emptyset \right\} \\ &= \bigwedge \left\{ \bigvee (cm^C_{pq}) \mid \forall cx_p, cx_q \in CU, cm^C_{pq} \neq \emptyset \right\} \\ &= \mathcal{F}(\mathbf{M}^C_{CDT}).\end{aligned}$$

□

Theorem 5.3 indicates that the discernibility function of a decision table is identical with its compacted version. Thus, the reducts derived from a compacted decision table are the same as those derived from its original version.

According to the conclusions of Theorems 5.1-5.3, it is easy to see that the same discernibility functions can be obtained using a decision table and its compacted version with regard to the positive region, Shannon entropy, and complement entropy, respectively. Therefore, we can undoubtedly draw the conclusion that reducts obtained from a decision table are identical with those from its compacted version for the three senses mentioned above.

5.2. Change of a compacted decision table resulting from a change in object value

To aid in the following analyses, we suppose that $CDT = (CU, C \cup CD)$ is a compacted decision table from $DT = (U, C \cup \{d\})$, where $CU = \{cx_1, cx_2, \ldots, cx_u\}$ $(u = |U/C|)$ and $CD = \{cd_1, cd_2, \ldots, cd_s\}$ $(s = |U/\{d\}|)$. In addition, suppose that $CDT' = (CU', C \cup CD')$ is a compacted decision table evolved from CDT owing to an object x_v of DT changing into x'_v, where $CU' = \{cx'_1, cx'_2, \ldots, cx'_u\}$, and $CD' = \{cd'_1, cd'_2, \ldots, cd'_s\}$. For the object x'_v, we use $f_{CDT}(x'_v, C)$ to indicate the values of x'_v on the condition attribute set C, and $f(x'_v, d)$ to represent the decision value of x'_v.

By means of Definition 4.1 and the relationships among a changed object and the objects in a compacted decision table, we investigate the change in compacted decision table in the following cases:

(1) $|[x_v]_C| = 1$ and $|[x'_v]_C| = 1$
In this case, because $\exists cx_p \in CU$ such that $f_{DT}(x_v, C) = f_{CDT}(cx_p, C)$ and $obj(cx_p) = 1$, and $f_{DT'}(x'_v, C) \neq f_{CDT}(cx_i, C)$ for $\forall cx_i \in CU$, it is easy to know that $|CU'| = |CU| = u$. Without a loss of generality, we suppose that $f_{CDT'}(cx'_p, C) = f_{DT'}(x'_v, C)$, $f_{CDT'}(cx'_p, CD') = f_{CDT}(cx_p, CD)$, $f_{CDT'}(cx'_j, C) = f_{CDT}(cx_j, C) (1 \leq j \leq u, j \neq p)$, $f_{CDT'}(cx'_j, CD') = f_{CDT}(cx_j, CD) (1 \leq j \leq u, j \neq p)$.

(2) $|[x_v]_C| = 1$ and $|[x'_v]_C| > 1$
In this case, because $\exists cx_p \in CU$ such that $f_{DT}(x_v, C) = f_{CDT}(cx_p, C)$ and $obj(cx_p) = 1$, and $\exists cx_q \in CU$ such that $f_{DT}(x'_v, C) = f_{CDT}(cx_q, C)$, it is easy to see that $|CU'| = |CU| = u - 1$. Without a loss of generality, we suppose that $f_{CDT'}(cx'_q, C) = f_{CDT}(cx_q, C)$, $f(x_v, d) = v_{d_r}$, $f_{CDT'}(cx'_q, cd'_k) = f_{CDT}(cx_q,\ cd_k) (1 \leq k \leq s,\ k \neq r)$, $f_{CDT'}(cx'_q, cd'_r) = f_{CDT}(cx_q, cd_r) \bigcup \{x_v\}$, and $f_{CDT'}(cx'_j, C) = f_{CDT}(cx_j, C) (1 \leq j \leq u, j \neq p, q)$, $f_{CDT'}(cx'_j, CD') = f_{CDT}(cx_j, CD) (1 \leq j \leq u, j \neq p, q)$.

(3) $|[x_v]_C| > 1$ and $|[x'_v]_C| = 1$
In this case, because $\exists cx_p \in CU$ such that $f_{DT}(x_v, C) = f_{CDT}(cx_p, C)$ and $obj(cx_p) > 1$, and $f_{DT'}(x'_v, C) \neq f_{CDT}(cx_i, C)$ for $\forall cx_i \in CU$, it is easy to see that $|CU'| = |CU| = u + 1$. Without a loss of generality, we suppose that $f_{CDT'}(cx'_{u+1}, C) = f_{DT'}(x'_v, C)$, $f(x_v, d) = v_{d_r}$, $f_{CDT'}(cx'_{u+1}, cd_r) = \{x_v\}$, $f_{CDT'}(cx'_{u+1}, cd_k) = \emptyset (1 \leq k \leq s, k \neq r)$, $f_{CDT'}(cx'_p, C) = f_{CDT}(cx_p, C)$, $f_{CDT'}(cx'_p, cd'_r) = f_{CDT}(cx_p, cd_r) - \{x_v\}$, $f_{CDT'}(cx'_p, cd_k) = f_{CDT}(cx_p, cd_k) (1 \leq$

$k \le s, k \ne r)$, $f_{CDT'}(cx'_i, C) = f_{CDT}(cx_i, C)(1 \le i \le u, i \ne p)$, $f_{CDT'}(cx'_i, CD') = f_{CDT}(cx_i, CD)(1 \le i \le u, i \ne p)$.

(4) $|[x_v]_C| > 1$ and $|[x'_v]_C| > 1$
In this case, because $\exists cx_p \in CU$ such that $f_{DT}(x_v, C) = f_{CDT}(cx_p, C)$ and $obj(cx_p) > 1$, and $\exists cx_q \in CU$ such that $f_{DT'}(x'_v, C) = f_{CDT}(cx_q, C)$, it is easy to see that $|CU'| = |CU| = u$. Without a loss of generality, we suppose that $f_{CDT'}(cx'_p, C) = f_{CDT}(cx_p, C)$, $f(x_v, d) = v_{d_r}$, $f_{CDT'}(cx'_p, cd_r) = f_{CDT}(cx_p, cd_r) - \{x_v\}$, $f_{CDT'}(cx'_p, cd_k) = f_{CDT}(cx_p, cd_k)(1 \le k \le s, k \ne r)$, $f_{CDT'}(cx'_q, C) = f_{CDT}(cx_q, C)$, $f_{CDT'}(cx'_q, cd_r) = f_{CDT}(cx_q, cd_r) \cup \{x_v\}$, $f_{CDT'}(cx'_q, cd_k) = f_{CDT}(cx_q, cd_k)(1 \le k \le s, k \ne r)$, $f_{CDT'}(cx'_i, C) = f_{CDT}(cx_i, C)(1 \le i \le u, i \ne p, q)$, $f_{CDT'}(cx'_i, CD') = f_{CDT}(cx_i, CD)(1 \le i \le u, i \ne p, q)$.

5.3. Relationship between reducts obtained by CDT' and DT'

In this section, we emphasize the relationship of reducts obtained through an updated compacted decision table (CDT') and an updated decision table (DT'), which can verify the effectiveness of these proposed discernibility matrices. In Section 5.1, we demonstrate that the reducts acquired from a compacted decision table (CDT) are identical with those acquired from its original version (DT). We can leverage this conclusion if an updated compacted decision table (CDT') and a compacted updated decision table ($DT'C$) can be proven to be the same. Thus, we first analyze the relationship between an updated compacted decision table (CDT') and a compacted updated decision table ($DT'C$) through the following theorem.

Theorem 5.4. *Given a decision table $DT = \{U, C \cup \{d\}\}$ and its compacted version $CDT = \{CU, C \cup CD\}$, $DT'C$ is identical to CDT', where $DT'C$ is a compacted table constructed by compacting DT', and DT' and CDT' are a decision table and a compacted decision table generated by changing the object x_v into x'_v, respectively.*

Proof. Suppose that $U/C = \cup_{i=1}^{u} X_p$, $x \in X_i$, $U/\{d\} = \cup_{i=1}^{s} Y_q, x \in Y_m$. There are four cases to be considered as follows:

(1) $|[x_v]_C| = 1$ and $|[x'_v]_C| = 1$
$\exists X_p \in U/C$ such that $x_v \in X_p$ and $|X_p| = 1$, and $x'_v \notin X_i$ for $\forall X_i \in U/C$, and it is clear that $|U/C| = |U'/C| = u$. By Definition 4.1, in $DT'C$, without a loss of generality, we suppose that $f_{DT'C}(x'c_p, C) = f_{DT'}(x'_v, C)$ and $f_{DT'C}(x'c_p, D'C) = f_{CDT}(cx_p, CD)$, and that $f_{DT'C}(x'c_i, C) = f_{DT}(X_i, C)(1 \le i \le u, i \ne p)$ and $f_{DT'C}(x'c_i, d'c_k) = f_{CDT}(cx_i, CD)(1 \le i \le u, i \ne p)$. Combined with Case (1) in Section 5.2, it is easy to see that $f_{DT'C}(x'c_p, C) = f_{CDT'}(cx'_p, C)$ and $f_{DT'C}(x'c_p, D'C) = f_{CDT'}(cx'_p, CD')$, and $f_{DT'C}(x'c_i, C) = f_{CDT'}(cx'_i, C)$ and $f_{DT'C}(x'c_i, D'C) = f_{CDT'}(cx'_i, CD')(1 \le i \le u, i \ne p)$.

(2) $|[x_v]_C| = 1$ and $|[x'_v]_C| > 1$
$\exists X_p \in U/C$ such that $x_v \in X_p$ and $|X_p| = 1$, and $\exists X_q \in U/C$ such that $x'_v \in X_q (p \ne q)$, and it is easy to see that $|U'/C| = u - 1$. By Definition 4.1, in $DT'C$, without a loss of generality, we suppose that $f_{DT'C}(x'c_q, C) = f_{DT}(X_q, C)$, $f_{DT}(x_v, d) = v_{d_r}$, $f_{DT'C}(x'c_q, d'c_r) = \{x \mid f_{DT}(x, d) = v_{d_r}, x \in X_q\} \cup \{x_v\}$, and $f_{DT'C}(x'c_q, d'c_k) = \{x \mid f_{DT}(x, d) = v_{d_k}, x \in X_q\}(1 \le k \le s, k \ne r)$; in addition, $f_{DT'C}(x'c_i, C) = f_{DT}(X_i, C)$, and $f_{DT'C}(x'c_i, d'c_k) = f_{CDT}(cx_i, CD)(1 \le i \le u, i \ne p, q)$. Combined with Case (2) in Section 5.2, it is easy to see that $f_{DT'C}(x'c_q, C) = f_{CDT'}(cx'_q, C)$ and $f_{DT'C}(x'c_q, D'C) = f_{CDT'}(cx'_q, CD')$, and $f_{DT'C}(x'c_i, C) = f_{CDT'}(cx'_i, C)$ and $f_{DT'C}(x'c_i, D'C) = f_{CDT'}(cx'_i, CD')(1 \le i \le u, i \ne p, q)$.

(3) $|[x_v]_C| > 1$ and $|[x'_v]_C| = 1$
$\exists X_p \in U/C$ such that $x_v \in X_p$ and $|X_p| > 1$, and $x'_v \notin X_i$ for $\forall X_i \in U/C$, and it is easy to see that $|U'/C| = u + 1$. By Definition 4.1, in $DT'C$, without a loss of generality, we suppose that $f_{DT'C}(x'c_{u+1}, C) = f_{DT'}(x'_v, C)$, $f_{DT}(x_v, d) = v_{d_r}$, $f_{DT'C}(x'c_{u+1}, d'c_r) = \{x_v\}$, $f_{DT'C}(x'c_{u+1}, d'c_k) = \emptyset(1 \le k \le s, k \ne r)$, $f_{DT'C}(x'c_p, C) = f_{DT}(X_p, C)$, $f_{DT'C}(x'c_p, d'c_r) = \{x \mid f_{DT}(x, d) = v_{d_r}, x \in (X_p)\} - \{x_v\}$, $f_{DT'C}(x'c_p, d'c_k) = \{x \mid f_{DT}(x, d) = v_{d_k}, x \in X_p\}(1 \le k \le s, k \ne r)$, $f_{DT'C}(x'c_i, C) = f_{DT}(X_i, C)(1 \le i \le u, i \ne p)$, and $f_{DT'C}(x'c_p, d'c_k) = \{x \mid f_{DT}(x, d) = v_{d_k}, x \in X_i\}(1 \le i \le u, i \ne p, 1 \le k \le s)$.
Combined with Case (3) in Section 5.2, it is easy to see that $f_{DT'C}(x'c_{u+1}, C) = f_{CDT'}(cx'_{u+1}, C)$ and $f_{DT'C}(x'c_{u+1}, D'C) = f_{CDT'}(cx'_{u+1}, CD')$, $f_{DT'C}(x'c_p, C) = f_{CDT'}(cx'_p, C)$ and $f_{DT'C}(x'c_p, D'C) = f_{CDT'}(cx'_p, CD')$, and $f_{DT'C}(x'c_i, C) = f_{CDT'}(cx'_i, C)$ and $f_{DT'C}(x'c_i, D'C) = f_{CDT'}(cx'_i, CD')(1 \le i \le u, i \ne p)$.

(4) $|[x_v]_C| > 1$ and $|[x'_v]_C| > 1$
$\exists X_p \in U/C$ such that $x_v \in X_p$ and $|X_p| > 1$, and $\exists X_q \in U/C$ such that $x'_v \in X_q (p \ne q)$, and it is easy to know that $|U'/C| = u$. By Definition 4.1, in $DT'C$, without a loss of generality, $f_{DT'C}(x'c_p, C) = f_{DT}(X_p, C)$, $f_{DT}(x_v, d) = v_{d_r}$, $f_{DT'C}(x'c_p, d'c_r) = \{x \mid f_{DT}(x, d) = v_{d_r}, x \in (X_p)\} - \{x_v\}$, $f_{DT'C}(x'c_p, d'c_k) = \{x \mid f_{DT}(x, d) = v_{d_k}, x \in X_p\}(1 \le k \le s, k \ne r)$, $f_{DT'C}(x'c_q, C) = f_{DT}(X_q, C)$, $f_{DT'C}(x'c_q, d'c_r) = \{x \mid f_{DT}(x, d) = v_{d_r}, x \in X_q\} \cup \{x_v\}$, $f_{DT'C}(x'c_q, d'c_k) = \{x \mid f_{DT}(x, d) = v_{d_k}, x \in X_q\}(1 \le k \le s, k \ne r)$, $f_{DT'C}(x'c_i, C) = f_{DT}(X_i, C)(1 \le i \le u, i \ne p)$, and $f_{DT'C}(x'c_p, d'c_k) = \{x \mid f_{DT}(x, d) = v_{d_k}, x \in X_i\}(1 \le i \le u, i \ne p, 1 \le k \le s)$.
Combined with Case (4) in Section 5.2, it is easy to see $f_{DT'C}(x'c_p, C) = f_{CDT'}(cx'_p, C)$ and $f_{DT'C}(x'c_p, D'C) = f_{CDT'}(cx'_p, CD')$, and $f_{DT'C}(x'c_q, C) = f_{CDT'}(cx'_q, C)$ and $f_{DT'C}(x'c_q, D'C) = f_{CDT'}(cx'_q, CD')$, and $f_{DT'C}(x'c_i, C) = f_{CDT'}(cx'_i, C)$ and $f_{DT'C}(x'c_i, D'C) = f_{CDT'}(cx'_i, CD')(1 \le i \le u, i \ne p, q)$.
Based on the analyses above, we draw the conclusion that the compacted decision table $DT'C$ is the same as the compacted decision table CDT'.

□

Based on the conclusion of Theorems 5.1-5.4, we can introduce the following significant corollary.

Corollary 5.1. *Given a decision table $DT = \{U, C \cup \{d\}\}$ and its compacted version $CDT = \{CU, C \cup CD\}$, if DT' is a decision table generated by changing an object x into x', and CDT' is a compacted decision table generated by varying the object x into x', then*

$$\mathcal{F}(\mathbf{M}^P_{DT'}) = \mathcal{F}(\mathbf{M}^P_{CDT'}), \mathcal{F}(\mathbf{M}^S_{DT'}) = \mathcal{F}(\mathbf{M}^S_{CDT'}), \mathcal{F}(\mathbf{M}^C_{DT'}) = \mathcal{F}(\mathbf{M}^C_{CDT'}).$$

Proof. Based on Theorem 5.4, it is easy to obtain $\mathcal{F}(\mathbf{M}^P_{DT'C}) = \mathcal{F}(\mathbf{M}^P_{CDT'})$, $\mathcal{F}(\mathbf{M}^S_{DT'C}) = \mathcal{F}(\mathbf{M}^S_{CDT'})$, and $\mathcal{F}(\mathbf{M}^C_{DT'C}) = \mathcal{F}(\mathbf{M}^C_{CDT'})$. Furthermore, by Theorem 5.1, we can conclude that $\mathcal{F}(\mathbf{M}^P_{DT'}) = \mathcal{F}(\mathbf{M}^P_{CDT'})$, by Theorem 5.2, we can conclude that $\mathcal{F}(\mathbf{M}^S_{DT'}) = \mathcal{F}(\mathbf{M}^S_{CDT'})$, and by Theorem 5.3, we can conclude that $\mathcal{F}(\mathbf{M}^C_{DT'}) = \mathcal{F}(\mathbf{M}^C_{CDT'})$. □

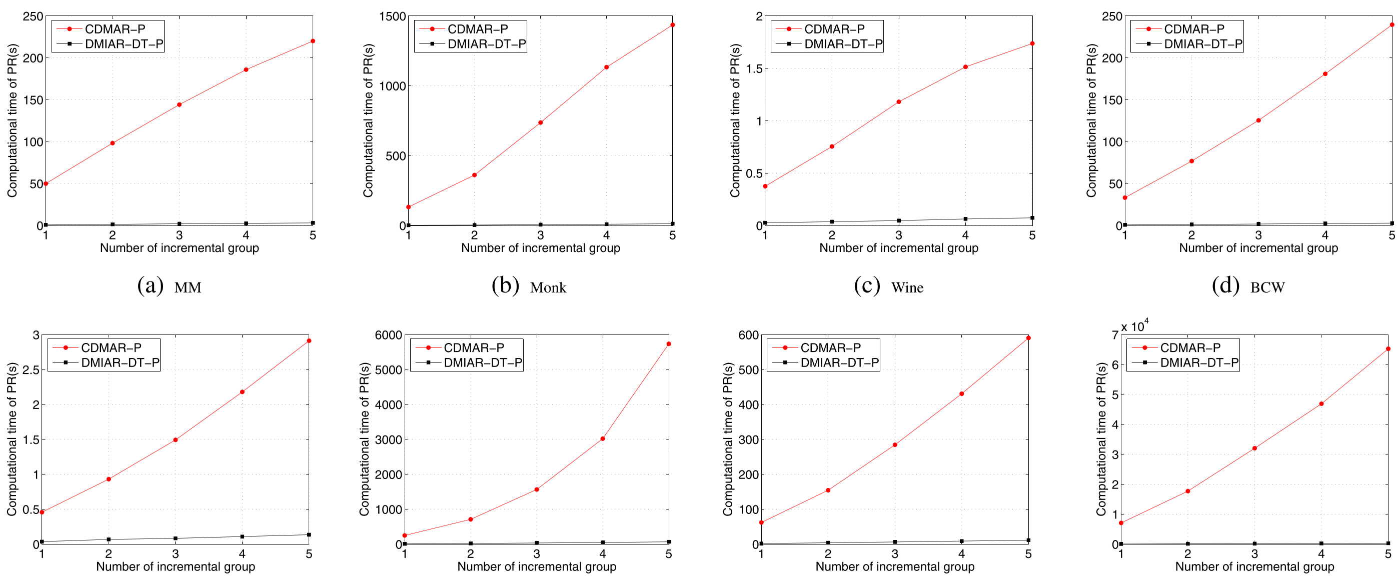
Corollary 5.1 indicates that when the values of the objects change, the discernibility function of table DT' evolved from a decision table is identical with that of CDT' from the same decision table in terms of the positive region, Shannon entropy, and complement entropy. From Corollary 5.1, we can draw the conclusion that the same reducts can be obtained from a decision table as from a compacted table when some values of the objects vary over time.

6. Experimental analysis

The following experiments were conducted to show the effectiveness of our proposed algorithm for datasets in which object values change over time. In these experiments, eight datasets were

Table 5
Datasets used in experiments.

ID	Datasets	Abbreviation	Attributes	Objects		
				Consistent part	Inconsistent part	Total
1	Mammographic Mass	MM	6	681	149	830
2	Monk's Problem	Monk	7	384	1327	1711
3	Wine	Wine	13	178	0	178
4	Breast Cancer Wisconsin(Original)	BCW	9	683	0	683
5	Spect	Spect	22	218	49	267
6	Spambase	Spam	57	625	3976	4601
7	Wine quality	WQ	32	163	4735	4898
8	Gesture phase	GP	18	1095	8806	9901



(a) MM (b) Monk (c) Wine (d) BCW (e) Spect (f) Spam (g) WQ (h) GP

Fig. 1. A comparison of the time taken for CDMAR-P and DMIAR-DT-P.

downloaded from the UCI Machine Learning Database Repository. All experiments were carried out on a personal computer with an Intel(R) 3.4 GHz Core(TM) i7-2600 and 4 GB of memory. The software used is Microsoft Visual 2013, and the programming language is C#.

To illustrate the efficiency of our proposed algorithms, we select 10%, 20%, 30%, 40%, and 50% as the objects of these datasets in Table 5, and replace these objects with new ones in which the value of each attribute is randomly selected from the attribute domain or assigned a new value. For each dataset after each variation (from 10% to 50%), a classical discernibility matrix based non-incremental attribute reduction algorithm (CDMAR), discernibility matrix based incremental attribute reduction algorithm for a decision table (DMIAR-DT), and discernibility matrix based incremental attribute reduction algorithm for a compacted decision table (DMIAR-CDT) were employed to compute their reducts.

The elapsed time of these algorithms was used to evaluate their performance from the perspective of efficiency, and Figs. 1, 2, 3 show the elapsed times of the CDMAR-P and DMIAR-DT-P algorithms, CDMAR-S and DMIAR-DT-S algorithms, and CDMAR-C and DMIAR-DT-C algorithms. In Fig. 1, we can see that the running time of DMIAR-DT-P was much less than that of CDMAR-P for all UCI datasets in Table 5. The experimental results illustrate that our proposed algorithm, DMIAR-CDT-P, is more efficient than CDMAR-P. Figs. 2 and 3 display similar results to those in Fig. 1, which indicates that DMIAR-DT-S and DMIAR-DT-C were also faster than CDMAR-S and CDMAR-C, respectively. In addition, Figs. 4, 5, 6 show the elapsed times of DMIAR-DT-P and DMIAR-CDT-P, DMIAR-DT-S and DMIAR-CDT-S, and DMIAR-DT-C and DMIAR-CDT-C. As shown in Fig. 4, we can see that the running time of DMIAR-CDT-P was much less than that of DMIAR-DT-P on all UCI datasets in Table 5. The experimental results illustrate that our proposed algorithm DMIAR-CDT-P was more efficient than DMIAR-DT-P. Figs. 5 and 6 show similar results to those in Fig. 4, and indicates that DMIAR-CDT-S and DMIAR-CDT-C are also faster than DMIAR-DT-S and DMIAR-DT-C, respectively. It is worth noting that the performances of these proposed algorithms improved as the number of changed objects increased, that is, the more the dataset changed, the more efficient these algorithms, DMIAR-CDT-P, DMIAR-CDT-S, DMIAR-CDT-C, DMIAR-DT-P, DMIAR-DT-S, and DMIAR-DT-C, were.

From the experimental analysis mentioned above, we determined that these incremental algorithms based on a compacted decision table are much faster than those based on the original version. To further clarify the reason for this acceleration, we analyzed the non-empty entries in the discernibility matrices of a decision table and the compacted version. From Tables 6, 7 and 8, it is easy to see that the numbers of non-empty entries in the discernibility matrices of each dataset significantly decrease after compaction, which illustrates the reason why incremental algorithms based a compacted table are more efficient. It should be pointed out that the performance of these proposed algorithms is closely related to the ratio of the numbers of non-empty entries

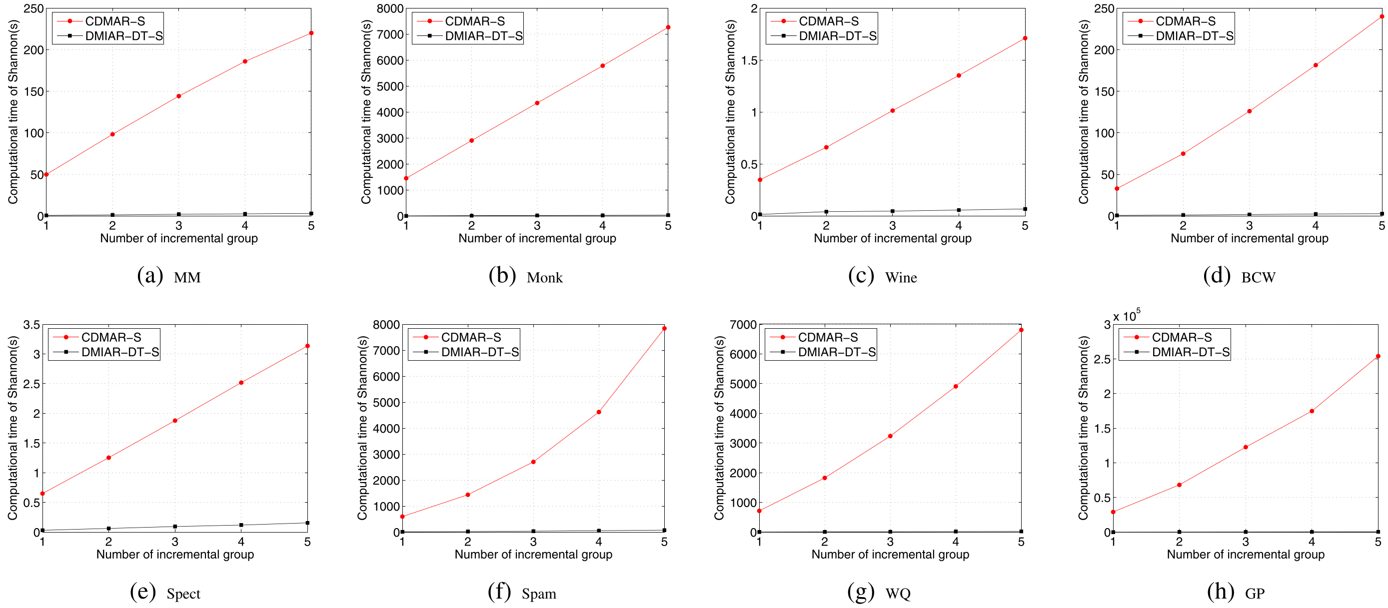


Fig. 2. A comparison of the time taken for CDMAR-S and DMIAR-DT-S.

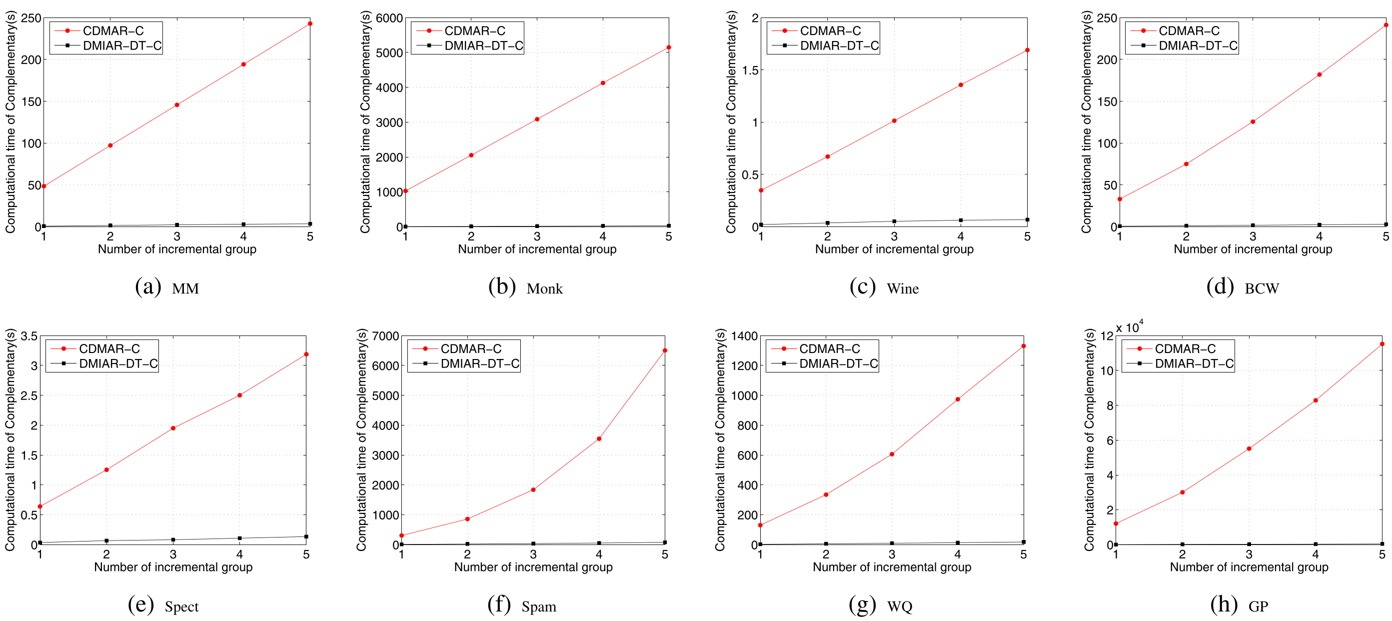


Fig. 3. A comparison of the time taken CDMAR-C and DMIAR-DT-C.

in the discernibility matrices of a dataset to the numbers of non-empty entries in the discernibility matrices of its compacted version. Take Table 5 and Fig. 4 as examples. In Table 5, for the dataset “Spect,” the ratios in all types of updating situations (10%, 20%, 30%, 40% and 50%) are 88.60%, 81.83%, 83.98%, 83.76% and 81.22%, which are much higher than those of the ”Spam” dataset, i.e., 9.76%, 16.69%, 13.12%, 10.75% and 8.62%, respectively. The results are consistent with those in Fig. 4 (e) and (f), i.e., the difference between the black and red lines in Fig. 4 (f) is also clearly bigger than that in Fig. 4 (e). Similar results can be found in Table 6 and Fig. 5, and in Table 7 and Fig. 6. Note that, essentially, the ratio of the numbers of non-empty entries in the discernibility matrices of a dataset to the those of non-empty entries in the discernibility matrices of its compacted version depends largely on the comparison ratio of a decision table, which is the ratio of objects in a decision table to those in its compacted version [40].

Furthermore, the difference in performance between these incremental attribute reduction algorithms caused by the different discernibility matrices (discernibility matrices in terms of the positive region, Shannon entropy, and complement entropy) of a compacted decision table need to be analyzed. Among these discernibility matrices, only the one regarding Shannon entropy has to spend extra time in comparing the decision distribution determined by each of two different objects in the inconsistent part of a compacted decision table (See Definition 4.3). Compared with the discernibility matrices in terms of Shannon entropy, the one regarding a positive region does not consider the differences among the objects in the inconsistent part of a compacted decision table

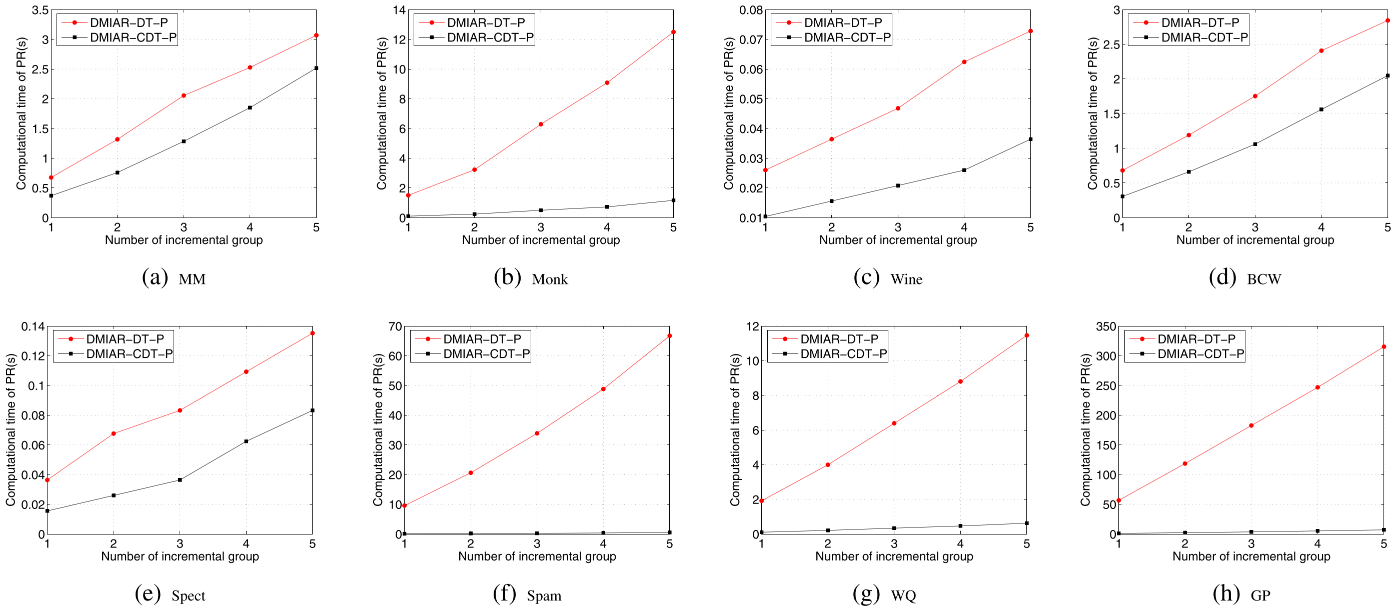


Fig. 4. A comparison of the time taken DMIAR-DT-P and DMIAR-CDT-P.

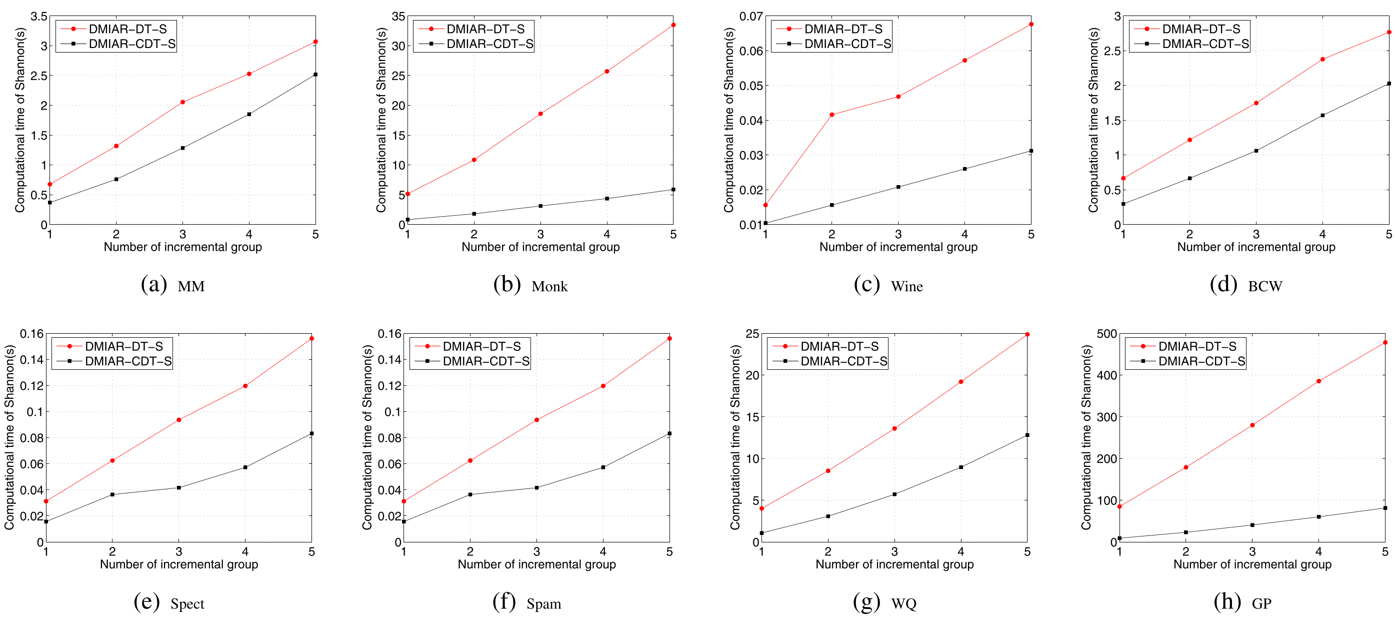


Fig. 5. A comparison of the time taken DMIAR-DT-S and DMIAR-CDT-S.

(See Definition 4.2), and the one with regard to complement entropy only compute the difference among condition attributes of each of two objects in the inconsistent part (See Definition 4.4). Consequently, an incremental algorithm for computing the reducts in terms of Shannon entropy can take much more time than those with regard to the positive region and complement entropy because the discernibility matrix of a compacted decision has to be iteratively updated based on the number of new objects.

To further illustrate this issue, we employ Tables 9 and 10. In Table 9, the 16 cases in Table 4, in which the decision distributions of each of two objects of an updated compacted decision table must be compared, are labelled with a star. The more cases a compacted decision covers during the updating process, the worse the incremental attribute reduction algorithm will perform on the compacted decision table. Table 10 shows the number of times each case appeared in the process of updating the eight datasets based five different percentages (10%, 20%, 30%, 40% and 50%). From Table 10, we can see that for the "WQ" dataset, the values of all percentages in the column "% of ⋆" are more than 95%, which indicates that more than 95% of the updated objects belong to one of $CT3$, $CT4$, $CT7$, $CT8$, $CT11$, $CT12$, $CT13$, $CT14$, $CT15$, and $CT16$. Thus, for the "WQ" dataset, the performance of the incremental attribute reduction algorithm in terms of Shannon entropy should be significantly worse than those of the positive region and complement entropy, which can be demonstrated through a comparison among Figs. 4(g), 5(g), and 6(g). This phenomenon is similar to the "WQ" dataset, and also appears in the "Monk," "Spam," and "GP" datasets; in other words, for "Monk," "Spam," and "GP", the perfor-

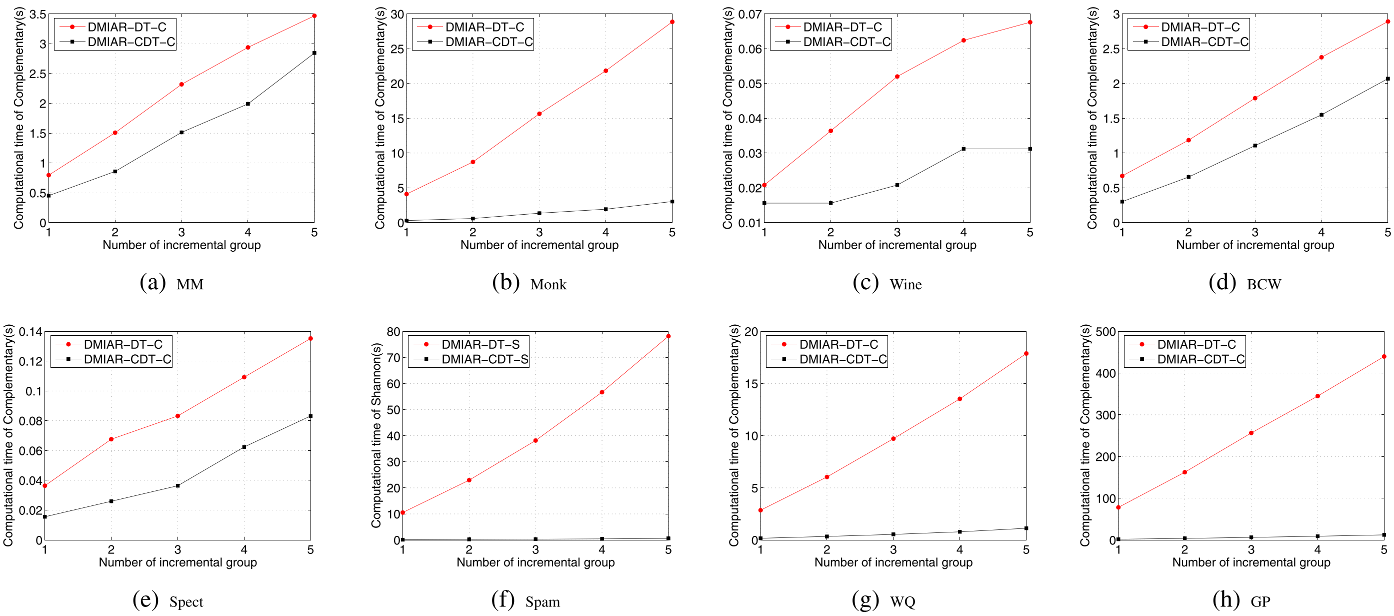


Fig. 6. A comparison of the time taken DMIAR-DT-C and DMIAR-CDT-C.

Table 6
Comparison of non-empty entries in discernibility matrices with regard to positive region.

Dataset	Algorithm	10%		20%		30%		40%		50%	
		Number	Ratio	Number	Ratio	Number	Ratio	Number	Ratio	Number	Ratio
MM	CDMAR-P	268189	100.00%	269723	100.00%	270766	100.00%	267982	100.00%	258822	100.00%
	DMIAR-DT-P	268189	100.00%	269723	100.00%	270766	100.00%	267982	100.00%	258822	100.00%
	DMIAR-CDT-P	142611	53.18%	149428	55.40%	154608	57.10%	157645	58.83%	157286	60.77%
Monk	CDMAR-P	299443	100.00%	378902	100.00%	481662	100.00%	524546	100.00%	534240	100.00%
	DMIAR-DT-P	299443	100.00%	378902	100.00%	481662	100.00%	524546	100.00%	534240	100.00%
	DMIAR-CDT-P	101959	34.05%	140432	37.06%	170132	35.32%	180602	34.43%	182362	34.13%
Wine	CDMAR-P	22488	100.00%	23240	100.00%	23378	100.00%	22594	100.00%	20814	100.00%
	DMIAR-DT-P	22488	100.00%	23240	100.00%	23378	100.00%	22594	100.00%	20814	100.00%
	DMIAR-CDT-P	11290	50.20%	12604	54.23%	13020	55.69%	12978	57.44%	12856	61.77%
BCW	CDMAR-P	244240	100.00%	268454	100.00%	287044	100.00%	297348	100.00%	300174	100.00%
	DMIAR-DT-P	244240	100.00%	268454	100.00%	287044	100.00%	297348	100.00%	300174	100.00%
	DMIAR-CDT-P	124300	50.89%	135192	50.36%	142632	49.69%	147432	49.58%	148910	49.61%
Spect	CDMAR-P	13740	100.00%	15321	100.00%	16782	100.00%	18142	100.00%	19592	100.00%
	DMIAR-DT-P	13740	100.00%	15321	100.00%	16782	100.00%	18142	100.00%	19592	100.00%
	DMIAR-CDT-P	12174	88.60%	12537	81.83%	14094	83.98%	15195	83.76%	15912	81.22%
Spam	CDMAR-P	212906	100.00%	304720	100.00%	435978	100.00%	571208	100.00%	768487	100.00%
	DMIAR-DT-P	212906	100.00%	304720	100.00%	435978	100.00%	571208	100.00%	768487	100.00%
	DMIAR-CDT-P	42063	19.76%	50868	16.69%	57212	13.12%	61400	10.75%	66274	8.62%
WQ	CDMAR-P	77040	100.00%	97498	100.00%	116282	100.00%	128578	100.00%	141939	100.00%
	DMIAR-DT-P	77040	100.00%	97498	100.00%	116282	100.00%	128578	100.00%	141939	100.00%
	DMIAR-CDT-P	53696	69.70%	67834	69.57%	81204	69.83%	88834	69.09%	96118	67.72%
GP	CDMAR-P	1516436	100.00%	1810754	100.00%	2033688	100.00%	2209351	100.00%	2451322	100.00%
	DMIAR-DT-P	1516436	100.00%	1810754	100.00%	2033688	100.00%	2209351	100.00%	2451322	100.00%
	DMIAR-CDT-P	360153	23.75%	427320	23.59%	465526	22.89%	493546	22.34%	503692	20.55%

mance of the incremental attribute reduction algorithm in terms of Shannon entropy is also worse than that of the positive region and complement entropy.

7. Conclusion

Because dynamic data extensively exist in real applications, attribute reduction for particular types of data has become a challenging issue in the field of rough sets. In this paper, we first introduced a discernibility matrix based incremental attribute reduction algorithm to incrementally compute all reducts, including the optimal reduct, of dynamic data. To further enhance the efficiency of the discernibility matrix based incremental attribute reduction algorithm, we defined a new compacted decision table, and based on a discernibility matrix of the compacted decision table, we developed an incremental attribute reduction algorithm. Some theorems and extensive experiments show that the incremental algorithm requires much less time to find the reducts than the former algorithm, and that the same reducts can be acquired by both. It should be pointed out that the incremental mechanisms of the discernibility matrices proposed in this paper are applicable to a case in which the objects in dynamic data are varied one by one. However, in real applications, the objects of dynamic data may vary in groups. To deal with dynamic data, we have to repetitively run these incremental algorithms based on their mechanism according

Table 7
Comparison of non-empty entries in discernibility matrices with regard to Shannon entropy.

Dataset	Algorithm	10%		20%		30%		40%		50%	
		Number	Ratio	Number	Ratio	Number	Ratio	Number	Ratio	Number	Ratio
MM	CDMAR-S	299328	100.00%	298086	100.00%	293410	100.00%	289062	100.00%	277424	100.00%
	DMIAR-DT-S	299328	100.00%	298086	100.00%	293410	100.00%	289062	100.00%	277424	100.00%
	DMIAR-CDT-S	165236	55.2%	170498	57.20%	171734	58.53%	173882	60.15%	171884	61.96%
Monk	CDMAR-S	560052	100.00%	664178	100.00%	781556	100.00%	829148	100.00%	829714	100.00%
	DMIAR-DT-S	560052	100.00%	664178	100.00%	781556	100.00%	829148	100.00%	829714	100.00%
	DMIAR-CDT-S	264932	47.30%	319346	48.08%	348604	44.60%	356504	43.00%	348178	41.96%
Wine	CDMAR-S	22488	100.00%	23240	100.00%	23378	100.00%	22594	100.00%	20814	100.00%
	DMIAR-DT-S	22488	100.00%	23240	100.00%	23378	100.00%	22594	100.00%	20814	100.00%
	DMIAR-CDT-S	11290	50.20%	12604	54.23%	13020	55.69%	12978	57.44%	12856	61.77%
BCW	CDMAR-S	244240	100.00%	268454	100.00%	287044	100.00%	297348	100.00%	300174	100.00%
	DMIAR-DT-S	244240	100.00%	268454	100.00%	287044	100.00%	297348	100.00%	300174	100.00%
	DMIAR-CDT-S	124300	50.89%	135192	50.36%	142632	49.69%	147432	49.58%	148910	49.61%
Spect	CDMAR-S	16018	100.00%	17422	100.00%	18222	100.00%	19360	100.00%	20584	100.00%
	DMIAR-DT-S	16018	100.00%	17422	100.00%	18222	100.00%	19360	100.00%	20584	100.00%
	DMIAR-CDT-S	14362	89.66%	14530	83.40%	15474	84.92%	16368	84.55%	16864	81.93%
Spam	CDMAR-S	241694	100.00%	344030	100.00%	483872	100.00%	628472	100.00%	825090	100.00%
	DMIAR-DT-S	241694	100.00%	344030	100.00%	483872	100.00%	628472	100.00%	825090	100.00%
	DMIAR-CDT-S	52404	21.68%	63322	18.41%	70510	14.57%	75284	11.98%	78702	9.54%
WQ	CDMAR-S	154598	100.00%	195830	100.00%	232734	100.00%	258278	100.00%	283764	100.00%
	DMIAR-DT-S	154598	100.00%	195830	100.00%	232734	100.00%	258278	100.00%	283764	100.00%
	DMIAR-CDT-S	121290	78.46%	153658	78.46%	182936	78.60%	201942	78.19%	218818	77.11%
GP	CDMAR-S	1970526	100.00%	2338854	100.00%	2633902	100.00%	2847730	100.00%	3120426	100.00%
	DMIAR-DT-S	1970526	100.00%	2338854	100.00%	2633902	100.00%	2847730	100.00%	3120426	100.00%
	DMIAR-CDT-S	594918	30.19%	701276	29.98%	774622	29.41%	820710	28.82%	834444	26.74%

Table 8
Comparison of non-empty entries in discernibility matrices with regard to complement entropy.

Dataset	Algorithm	10%		20%		30%		40%		50%	
		Number	Ratio	Number	Ratio	Number	Ratio	Number	Ratio	Number	Ratio
MM	CDMAR-C	299708	100.00%	293620	100.00%	293582	100.00%	289210	100.00%	277542	100.00%
	DMIAR-DT-C	299708	100.00%	293620	100.00%	293582	100.00%	289210	100.00%	277542	100.00%
	DMIAR-CDT-C	165616	55.26%	171960	57.24%	171906	58.55%	174030	60.17%	172002	61.97%
Monk	CDMAR-C	575550	100.00%	713108	100.00%	801534	100.00%	851158	100.00%	848970	100.00%
	DMIAR-DT-C	575550	100.00%	713108	100.00%	801534	100.00%	851158	100.00%	848970	100.00%
	DMIAR-CDT-C	280430	48.72%	335340	49.42%	368582	45.98%	378514	44.47%	367434	43.28%
Wine	CDMAR-C	22488	100.00%	23242	100.00%	23378	100.00%	22594	100.00%	20814	100.00%
	DMIAR-DT-C	22488	100.00%	23242	100.00%	23378	100.00%	22594	100.00%	20814	100.00%
	DMIAR-CDT-C	11290	50.20%	13466	54.23%	13020	55.69%	12978	57.44%	12856	61.77%
BCW	CDMAR-C	244240	100.00%	269444	100.00%	287044	100.00%	297348	100.00%	300174	100.00%
	DMIAR-DT-C	244240	100.00%	269444	100.00%	287044	100.00%	297348	100.00%	300174	100.00%
	DMIAR-CDT-C	124300	50.89%	135154	50.36%	142632	49.69%	147432	49.58%	148910	49.61%
Spect	CDMAR-C	16050	100.00%	16008	100.00%	18228	100.00%	19362	100.00%	20584	100.00%
	DMIAR-DT-C	16050	100.00%	16008	100.00%	18228	100.00%	19362	100.00%	20584	100.00%
	DMIAR-CDT-C	14394	89.68%	14824	83.43%	15480	84.92%	16370	84.55%	16864	81.93%
Spam	CDMAR-C	241748	100.00%	344104	100.00%	483966	100.00%	628556	100.00%	825124	100.00%
	DMIAR-DT-C	241748	100.00%	344104	100.00%	483966	100.00%	628556	100.00%	825124	100.00%
	DMIAR-CDT-C	52458	21.70%	63396	18.42%	70604	14.59%	75368	11.99%	78736	9.54%
WQ	CDMAR-C	154872	100.00%	196290	100.00%	233122	100.00%	258874	100.00%	284484	100.00%
	DMIAR-DT-C	154872	100.00%	196290	100.00%	233122	100.00%	258874	100.00%	284484	100.00%
	DMIAR-CDT-C	121564	78.49%	154118	78.51%	183324	78.64%	202538	78.24%	219538	77.17%
GP	CDMAR-C	1970934	100.00%	2339410	100.00%	2634506	100.00%	2848196	100.00%	3120970	100.00%
	DMIAR-DT-C	1970934	100.00%	2339410	100.00%	2634506	100.00%	2848196	100.00%	3120970	100.00%
	DMIAR-CDT-C	595326	30.21%	701832	30.00%	775226	29.43%	821176	28.83%	834988	26.75%

Table 9
The 16 cases in Table 4, in which the decision distributions of each of two objects must be compared.

	$obj(cx_q) = 0, \lvert\sigma_{CDT'}(cx'_q)\rvert = 1$	$\lvert\sigma_{CDT}(cx_q)\rvert = 1, \lvert\sigma_{CDT'}(cx'_q)\rvert = 1$	$\lvert\sigma_{CDT}(cx_q)\rvert = 1, \lvert\sigma_{CDT'}(cx'_q)\rvert > 1$	$\lvert\sigma_{CDT}(cx_q)\rvert > 1, \lvert\sigma_{CDT'}(cx'_q)\rvert > 1$
$\lvert\sigma_{CDT}(cx_p)\rvert = 1, obj(cx'_p) = 0$	CT1	CT2	CT3⋆	CT4⋆
$\lvert\sigma_{CDT}(cx_p)\rvert = 1, \lvert\sigma_{CDT'}(cx'_p)\rvert = 1$	CT5	CT6	CT7⋆	CT8⋆
$\lvert\sigma_{CDT}(cx_p)\rvert > 1, \lvert\sigma_{CDT'}(cx'_p)\rvert = 1$	CT9	CT10	CT11⋆	CT12⋆
$\lvert\sigma_{CDT}(cx_p)\rvert > 1, \lvert\sigma_{CDT'}(cx'_p)\rvert > 1$	CT13⋆	CT14⋆	CT15⋆	CT16⋆

Table 10
Number of times each case appeared during the process of updating the eight datasets based on five different percentages.

Dataset		CT1	CT2	CT3 ⋆	CT4 ⋆	CT5	CT6	CT7 ⋆	CT8 ⋆	CT9	CT10	CT11 ⋆	CT12 ⋆	CT13 ⋆	CT14 ⋆	CT15 ⋆	CT16 ⋆	Total	% of ⋆
MM	10%	28	4	1	0	15	19	1	2	2	2	0	0	5	4	0	0	83	15.66%
	20%	54	10	2	0	29	34	1	5	5	3	3	1	8	8	0	3	166	18.67%
	30%	82	17	2	5	39	43	2	12	13	4	3	2	12	8	1	4	249	20.48%
	40%	107	28	3	9	57	46	2	19	15	5	6	2	15	10	1	7	332	22.29%
	50%	139	34	5	14	69	56	2	24	15	8	7	4	17	13	1	7	415	22.65%
Monk	10%	0	0	0	0	27	8	1	1	19	3	2	1	86	16	4	3	171	66.67%
	20%	0	0	0	0	49	28	8	7	26	12	6	5	126	51	11	13	342	66.37%
	30%	1	1	1	2	59	53	13	9	38	27	14	10	151	87	20	27	513	65.11%
	40%	1	2	2	5	69	83	19	24	41	44	21	17	160	125	32	39	684	64.91%
	50%	1	9	4	10	75	108	28	43	44	66	30	27	161	160	39	50	855	64.56%
Wine	10%	13	0	0	0	4	1	0	0	0	0	0	0	0	0	0	0	18	0.00%
	20%	22	0	0	0	9	5	0	0	0	0	0	0	0	0	0	0	36	0.00%
	30%	33	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	53	0.00%
	40%	41	2	0	0	13	15	0	0	0	0	0	0	0	0	0	0	71	0.00%
	50%	49	2	0	0	15	23	0	0	0	0	0	0	0	0	0	0	89	0.00%
BCW	10%	39	1	0	0	17	11	0	0	0	0	0	0	0	0	0	0	68	0.00%
	20%	73	6	0	0	26	31	0	0	0	0	0	0	0	0	0	0	136	0.00%
	30%	112	10	0	0	32	51	0	0	0	0	0	0	0	0	0	0	205	0.00%
	40%	152	13	0	0	36	72	0	0	0	0	0	0	0	0	0	0	273	0.00%
	50%	193	15	0	0	39	94	0	0	0	0	0	0	0	0	0	0	341	0.00%
Spect	10%	19	0	0	0	1	0	0	0	0	0	0	0	3	3	1	0	27	25.93%
	20%	37	1	0	0	2	0	0	0	1	0	0	0	5	6	1	0	53	22.64%
	30%	58	1	0	0	4	0	0	0	4	0	0	0	6	6	1	0	80	16.25%
	40%	77	2	0	0	7	0	0	0	5	0	0	0	7	7	1	0	106	14.15%
	50%	97	3	0	0	9	0	0	0	6	0	0	0	8	9	1	0	133	13.53%
Spam	10%	21	1	0	0	41	17	0	0	2	0	0	0	25	190	1	162	460	82.17%
	20%	34	1	0	0	59	49	0	0	2	0	0	0	40	407	4	324	920	84.24%
	30%	44	2	0	0	75	75	0	0	3	0	0	0	48	637	5	491	1380	85.58%
	40%	56	4	0	0	86	109	0	0	4	0	0	1	54	856	7	663	1840	85.92%
	50%	71	7	0	0	95	143	0	1	8	1	0	2	58	1070	8	836	2300	85.87%
WQ	10%	7	0	0	0	6	2	0	0	4	0	0	0	111	34	61	265	490	96.12%
	20%	14	0	0	0	12	2	0	0	7	0	0	1	148	50	89	656	979	96.42%
	30%	20	1	0	0	15	3	0	0	13	0	1	1	177	65	113	1060	469	96.46%
	40%	29	1	0	0	20	5	2	2	15	1	3	2	192	75	129	1483	959	96.38%
	50%	38	3	2	0	22	5	2	3	17	3	4	3	210	92	143	1902	449	96.41%
GP	10%	18	2	0	0	69	24	0	1	6	1	0	0	187	149	80	453	990	87.88%
	20%	38	4	0	0	104	75	3	1	13	4	3	0	219	295	108	1113	1980	87.98%
	30%	57	10	1	0	128	127	4	11	18	5	5	1	244	393	135	1831	2970	88.38%
	40%	75	16	2	0	151	185	5	20	19	8	5	4	255	471	149	2595	3960	88.54%
	50%	90	28	5	2	162	261	5	25	23	10	8	8	265	561	157	3340	4950	88.40%

to the number of changing objects, which can significantly affect the performance of updating the reducts. Hence, developing incremental attribute reduction algorithms that can process all changed objects concurrently is a significant and imperative are of future study.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Nos. 61772323, 61303008, 61432011, 61402272 and U1435212), the National Key Basic Research and Development Program of China (973) (No. 2013CB329404), and the Scientific and Technological Innovation Programs of Higher Education Institutions in Shanxi Province, China (No. 2016111).

References

[1] D.G. Chen, C.Z. Wang, Q.H. Hu, A new approach to attribute reduction of consistent and inconsistent covering decision systems with covering rough sets, Inf. Sci. (Ny) 177 (17) (2007) 3500–3518.
[2] D.G. Chen, Y.Y. Yang, Z. Dong, An incremental algorithm for attribute reduction with variableprecision rough sets, Appl. Soft. Comput. 45 (2016) 129–149.
[3] S. Guan, S. Li, Incremental learning with respect to new incoming input attributes, Neural Process. Lett. 14 (3) (2001) 241–260.
[4] F. Hu, G.Y. Wang, H. Huang, Y. Wu, Incremental attribute reduction based on elementary sets, in: Proceeding of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, Regina, Canada, in: in: LNCS, volume 3641, 2005, pp. 185–193.
[5] Q.H. Hu, Z.X. Xie, D.R. Yu, Hybrid attribute reduction based on a novel fuzzy–rough model and information granulation, Pattern Recognit. 40 (12) (2007) 3509–3521.
[6] Q.H. Hu, J.F. Liu, D.R. Yu, Mixed feature selection based on granulation and approximation, Knowl. Based Syst. 21 (4) (2008) 294–304.
[7] Y.G. Jing, T.R. Li, J.F. Huang, An incremental attribute reduction approach based on knowledge granularity under the attribute generalization, Int. J. Approx. Reason. 76 (2016) 80–95.
[8] 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.
[9] 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.
[10] J.Y. Liang, K.S. Chin, C.Y. Dang, C.M. Yam, Richid, a new method for measuring uncertainty and fuzziness in rough set theory, Int. J. Gen. Syst. 31 (4) (2002) 331–342.
[11] J.Y. Liang, Z.Z. Shi, D.Y. Li, M.J. Wierman, The information entropy, rough entropy and knowledge granulation in incomplete information table, Int. J. Gen. Syst. 35 (6) (2006) 641–654.
[12] J.Y. Liang, W. Wei, Y.H. Qian, An incremental approach to computation of a core based on conditional entropy, Syst. Eng.-Theor. Pract. 28 (4) (2008) 81–89.
[13] J.Y. Liang, J.H. Wang, Y.H. Qian, A new measure of uncertainty based on knowledge granulation for rough sets, Inf. Sci. (Ny) 179 (4) (2009) 458–470.
[14] 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.
[15] D. Liu, T.R. Li, D. Ruan, W.L. Zou, An incremental approach for inducing knowledge from dynamic information systems, Fundam. Inf. 94 (2) (2009) 245–260.
[16] D. Liu, T.R. Li, D. Ruan, J.B. Zhang, Incremental learning optimization on knowledge discovery in dynamic business intelligent systems, J. Global Optim. 51 (2011) 325–344.
[17] D. Liu, T.R. Li, J.B. Zhang, A rough set-based incremental approach for learning knowledge in dynamic incomplete information systems, Int. J. Approx. Reason. 55 (2014) 1764–1786.
[18] G.L. Liu, Z. Hua, Z.H. Chen, A general reduction algorithm for relation decision systems and its applications, Knowl. Based Syst. 119 (2017) 87–93.

[19] Z.T. Liu, An incremental arithmetic for the smallest reduction of attributes, Chin. J. Electron. 27 (11) (1999) 96–98.
[20] J.S. Mi, W.Z. Wu, W.X. Zhang, Knowledge reducts based on variable precision rough set theory, Syst. Eng.-Theor. Pract. 24 (1) (2004) 76–82.
[21] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston, 1991.
[22] Z. Pawlak, A. Sknwron, Rough sets and boolean reasoning, Inf. Sci. (Ny) 177 (1) (2007) 41–73.
[23] Z. Pawlak, A. Skowron, Rudiments of rough sets, Inf. Sci. (Ny) 177 (1) (2007) 3–27.
[24] Y.H. Qian, J.Y. Liang, Combination entropy and combination granulation in incomplete information system, Lect. Notes Comput. Sci. 4062 (2006) 184–190.
[25] Y.H. Qian, J.Y. Liang, Combination entropy and combination granulation in rough set theory, Int. J. Uncertainty, Fuzziness Knowl.-Based Syst. 16 (2) (2008) 179–193.
[26] Y.H. Qian, J.Y. Liang, W. Pedrycz, et al., Positive approximation: an accelerator for attribute reduction in rough set theory, Artif. Intell. 174 (9) (2010) 597–618.
[27] C.E. Shannon, The mathematical theory of communication, Bell Syst. Techn. J. 27 (3, 4) (1948) 373–423.
[28] W.H. Shu, H. Shen, Updating attribute reduction in incomplete decision systems with the variation of attribute set, Int. J. Approx. Reason. 55 (2014) 867–884.
[29] W.H. Shu, H. Shen, Incremental feature selection based on rough set in dynamic incomplete data, Pattern Recognit. 47 (2014) 3890–3906.
[30] 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.
[31] R.W. Swiniarski, A. Skowron, Rough set methods in feature selection and recognition, Pattern Recognit. Lett. 24 (6) (2003) 833–849.
[32] S.H. Teng, M. Lu, A.F. Yang, J. Zhang, Y.J. Nian, M. He, Efficient attribute reduction from the viewpoint of discernibility, Inf. Sci. (Ny) 326 (2016) 297–314.
[33] C.Z. Wang, Y.L. Qi, M.W. Shao, Q.H. Hu, D.G. Chen, Y.H. Qian, Y.J. Lin, A fitting model for feature selection with fuzzy rough sets, IEEE Trans. Fuzzy Syst. 25 (4) (2016) 741–753.
[34] C.Z. Wang, M.W. Shao, Q. He, Y.H. Qian, Y.L. Qi, Feature subset selection based on fuzzy neighborhood rough sets, Knowl. Based Syst. 111 (1) (2016) 173–179.
[35] F. Wang, J.Y. Liang, C.Y. Dang, Attribute reduction for dynamic data sets, Appl. Soft Comput. 13 (2013) 676–689.
[36] F. Wang, J.Y. Liang, Y.H. Qian, Attribute reduction: a dimension incremental strategy, Knowl. Based Syst. 39 (2013) 95–108.
[37] G.Y. Wang, H. Yu, D.C. Yang, Decision table reduction based on conditional information entropy, Chin. J. Comput. 25 (7) (2002) 759–766.
[38] W. Wei, J.Y. Liang, Y.H. Qian, F. Wang, An attribute reduction approach and its accelerated version for hybrid data, IEEE ICCI (2009) 167–173.
[39] W. Wei, J.Y. Liang, J.H. Wang, Y.H. Qian, Decision-relative discernibility matrixes in the sense of entropies, Int. J. Gen. Syst. 42 (7) (2013) 721–738.
[40] W. Wei, J.H. Wang, J.Y. Liang, X. Mi, C.Y. Dang, Compacted decision tables based attribute reduction, Knowl.–Based Syst. 86 (2015) 261–277.
[41] W. Wei, J.B. Cui, J.Y. Liang, J.H. Wang, Fuzzy rough approximations for set-valued data, Inf. Sci. (Ny) 360 (2016) 181–201.
[42] J. Xie, X.F. Shen, H.F. Liu, Research on an incremental attribute reduction based on relative positive region, J. Comput. Inf. Syst. 9 (16) (2013) 6621–6628.
[43] Y.T. Xu, L.S. Wang, R.Y. Zhang, A dynamic attribute reduction algorithm based on 0–1 integer programming, Knowl. Based Syst. 24 (2011) 1341–1347.
[44] M. Yang, An incremental updating algorithm of the computation of a core based on the improved discernibility matrix, Chin. J. Comput. 29 (3) (2006) 407–413.
[45] M. Yang, An incremental updating algorithm for attribute reduction based on improved discernibility matrix, Chin. J. Comput. 30 (5) (2007) 815–822.
[46] Y.Y. Yao, Y. Zhao, Attribute reduction in decision-theoretic rough set models, Inf. Sci. (Ny) 178 (17) (2008) 3356–3373.
[47] Y.Y. Yao, Y. Zhao, Discernibility matrix simplification for constructing attribute reducts, Inf. Sci. (Ny) 179 (2009) 867–882.
[48] J.B. Zhang, T.R. Li, D. Ruan, D. Liu, Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems, Int. J. Approx. Reason. 53 (2012) 620–635.