Information Sciences

Simplified rough sets: Accelerating data analysis and knowledge discovery

--Manuscript Draft--

Manuscript Number:	INS-D-24-2563
Article Type:	Full length article
Keywords:	Attribute reduction; Computational efficiency; Rough sets; Upper and lower approximations

Abstract:	Rough set (RS) was first proposed by Z. Pawlak in 1982. For over forty years, a large number of RS models have been developed to solve various data problems. However, most RS models are designed based on inherent rules, and their mathematical structures are similar and complex. For this reason, the efficiency of RS methods in analyzing data has not been significantly improved. To address this issue, we propose some new rules to simplify traditional RS models. These simplified RS models, which are equivalent to traditional RS models, can mine data more quickly. In this paper, we take Pawlak RS as an example to compare the computational efficiency between the simplified Pawlak RS (SPRS) and the traditional RSs. Numerical experiments confirm that the computational efficiency of the SPRS is not only significantly higher than that of traditional Pawlak RS (TPRS) but even higher than that of most existing RS models. This indicates that RS models could have mined data more quickly, but the computational efficiency of RS method has been severely underestimated for a long time. Therefore, the research results of this article will greatly promote the development and application of RS theory.

Simplified rough sets: Accelerating data analysis and knowledge discovery

Qingzhao Kong[a], Conghao Yan[a], Weihua Xu[b,*]

[a]Department of Science, Jimei University, Xiamen, 361021, PR China
[b]College of Artificial Intelligence, Southwest University, Chongqing, 400715, PR China

Abstract

Rough set (RS) was first proposed by Z. Pawlak in 1982. For over forty years, a large number of RS models have been developed to solve various data problems. However, most RS models are designed based on inherent rules, and their mathematical structures are similar and complex. For this reason, the efficiency of RS methods in analyzing data has not been significantly improved. To address this issue, we propose some new rules to simplify traditional RS models. These simplified RS models, which are equivalent to traditional RS models, can mine data more quickly. In this paper, we take Pawlak RS as an example to compare the computational efficiency between the simplified Pawlak RS (SPRS) and the traditional RSs. Numerical experiments confirm that the computational efficiency of the SPRS is not only significantly higher than that of traditional Pawlak RS (TPRS) but even higher than that of most existing RS models. This indicates that RS models could have mined data more quickly, but the computational efficiency of RS method has been severely underestimated for a long time. Therefore, the research results of this article will greatly promote the development and application of RS theory.

Keywords: Attribute reduction, Computational efficiency, Rough sets, Upper and lower approximations.

1. Introduction

After experiencing the agricultural age, industrial age, and information age, humanity is entering the era of digital intelligence. In today′s society, data is more massive, complex, and important than ever before. Data has become one of the very important production factors. Quickly and accurately analyzing data has become an important issue of the times. This requires us to continuously explore various methods to handle complex and massive data.

1.1. Overview of related works

So far, many methods and theories have been proposed to deal with various types of data. For example, in the 20th century, with the discovery of maximum likelihood method, hypothesis testing method, trust inference method, and Bayesian decision theory, statistics rapidly developed and matured [1,2]. Based on the theory of fuzzy sets established by L.A. Zadeh [3], fuzzy theory is developed to solve fuzzy reasoning and fuzzy decision [4-6]. In addition, quotient space theory is explored to address complex data and achieve the goal of reducing computational complexity [7].

In 1982, Z. Pawlak proposed rough set (RS) model, which uses two precise sets, namely upper and lower approximations (ULAs), to approximate a set with fuzzy boundaries [8]. RS theory has undergone over 40 years of development, and has achieved many results in the establishment of system theory, computational models, and the development of application systems [9-12]. In RS theory, membership relationship is no longer a primitive concept, and there is no need to artificially assign a membership degree to an element, which effectively avoids the influence of subjective factors. When RS is used to analyze data, no prior knowledge is required and all parameters can be

*Corresponding author is W.H. Xu(E-mail: chxuwh@gmail.com).
Email addresses: kongqingzhao@163.com (Qingzhao Kong), 840223328@qq.com (Conghao Yan), chxuwh@gmail.com (Weihua Xu)

obtained from the sample set of the information table. Therefore, RS method extensively involves many fields such as knowledge representation and discovery, uncertain reasoning, granular computing, and feature selection [13-17].

Traditional Pawlak RS (TPRS) model, as we all know, is defined by an equivalence relationship or a partition on the universe. For any target concept X, the union of all equivalent classes contained in X is called the lower approximation (LA) of X. While the union of all equivalent classes whose intersections with X are not empty is referred to as the upper approximation (UA) of X. However, in most cases, it is not possible to induce an equivalence relationship or a partition from the universe. Therefore, many generalized RS models have been developed to address various data problems [18-21].

Meanwhile, we note that most of the mathematical structures of these generalized RS models are similar to that of the TPRS model. Therefore, the computational efficiency of most existing models is not significantly different from that of TPRS model.

1.2. Defects in existing rough sets

Although scholars are constantly trying to improve RS models in order to mine data more quickly, most of the proposed RS models are constructed based on the methods provided by Z. Pawlak, namely the Pawlak rules. That is to say, the mathematical structure of most RS models is similar to that of the TPRS model. This has resulted in little improvement in the computational efficiency of RS theory for over 40 years. There are two reasons why existing RS models cannot further analyze data quickly.

• For the RS models based on Pawlak rules, too much data needs to be analyzed. When RS method is employed to represent the target concept, all data in the data set, namely the universe, has to be examined. For example, in the TPRS model, the equivalent class of each data in the data set must be calculated. This is the main reason that restricts TPRS from quickly analyzing data.

• The process of calculating ULAs of the existing RS models is cumbersome, which reduces computational efficiency of RS models. For example, when the UA of TPRS is constructed, it is necessary to first calculate the equivalence classes of all data in the universe, and then verify whether the intersection of each equivalence class and the target concept is an empty set.

1.3. Our work

Based on the above analysis, one can find that in order to improve computational efficiency of RS models, we have to overcome the above two defects, especially the first one. Therefore, the motivation of this study is to propose some new rules and simplify existing RS models. And these simplified rough set models are not only equivalent to traditional models but also have higher computational efficiency. Here, taking Pawlak RS as an example, we will introduce simplified Pawlak RS (SPRS) that is equivalent to TPRS and can process data more efficiently. The main contributions of this article are listed as follows.

(1) Based on the new rules, the TPRS can be simplified and equivalently defined. Then, SPRS model has two characteristics: First, calculating the ULAs of SPRS involves less data. Second, the structure of the SPRS model is simpler. These advantages result in SPRS having higher computational efficiency than TPRS. For example, the time complexity of algorithms for calculating the ULAs of SPRS is linear in terms of the universe, while that of TPRS is quadratic.

(2) Although massive RS models have been introduced to solve various data problems, it is not difficult to see that most of these models can be equivalently redesigned based on these rules proposed in this paper. This means that with the support of these new rules, the computing efficiency of RS method can be generally improved. This will inevitably inject new vitality into the development and application of RS theory.

The remaining parts of the paper are arranged as follows. Some basic and important concepts in TPRS are listed in Section 2. In section 3, The SPRS model and other related results are presented in detail. In Section 4, we study the relationship between attribute reductions of UA and LA on Pawlak RS. It is proved that these two types of reduction are actually equivalent. Some algorithms on SPRS are developed in Section 5. In Section 6, some numerical experiments are designed. The computing efficiency of SPRS is analyzed. Experimental results show that SPRS is more effective in mining data than traditional RS models. In section 7, the main content of this article is briefly summarized.

2. Preliminaries

A sequence group

$$I = (U, AT, \{V_a | a \in AT\}, \{f_a | a \in AT\}) \tag{1}$$

is called an information table, where U, AT, V_a and f_a represent the universe, attribute set, attribute value of attribute a and information function about attribute a, respectively [22,23]. And $U/E_{AT} = \{[x]_{AT} | x \in U\}$ is the partition on U, where $[x]_{AT}$ is the equivalence class of x.

Since TPRS model was proposed, it has been extensively and deeply studied. Several equivalent definitions of TPRS have been listed as follows [8].

Definition 2.1 In an information table represented by Eq.(1), for each target concept $X \subseteq U$, the UA of X can be shown as follows.

$$\overline{apr}_{AT}(X) = \{x \in U \mid [x]_{AT} \cap X \neq \emptyset\} \tag{2}$$

$$= \cup \{[x]_{AT} \in U/E_{AT} \mid [x]_{AT} \cap X \neq \emptyset\}. \tag{3}$$

And the LA of X (where $X^C = U - X$) can be written as follows.

$$\underline{apr}_{AT}(X) = \{x \in U \mid [x]_{AT} \subseteq X\}, \tag{4}$$

$$= \cup \{[x]_{AT} \in U/E_{AT} \mid [x]_{AT} \subseteq X\} \tag{5}$$

$$= (\overline{apr}_{AT}(X^C))^C. \tag{6}$$

Eq.s (2)-(6) show classical methods for constructing TPRS. These methods involve two issues: One is that the equivalence class of each object needs to be calculated. The other is that the relationship between each equivalence class and the target concept has to be distinguished. Here, these classical methods are referred to as Pawlak rules. To address various learning tasks, more and more generalized RS models are developed. However, most of these models are designed by Pawlak rules. Therefore, computational efficiency of these models is similar to that of TPRS model. Based on TPRS model, for any target concept $X \subseteq U$, all data of the information table can be divided into three disjoint parts, namely positive, negative and boundary regions, as follows.

$$Pos_{AT}(X) = \underline{apr}_{AT}(X);$$
$$Neg_{AT}(X) = U - \overline{apr}_{AT}(X);$$
$$Bou_{AT}(X) = \overline{apr}_{AT}(X) - \underline{apr}_{AT}(X).$$

Attribute reduction, also known as feature selection, is a core issue of RS theory. In order to achieve various learning tasks, many types of attribute reduction are proposed [8,24-30]. In what follows, two widely used and important kinds of attribute reduction are shown as follows.

Definition 2.2 In an information table represented by Eq.(1), for any target concept $X \subseteq U$, if $A \subseteq AT$ satisfies the following two conditions:

(1) $\underline{apr}_{A}(X) = \underline{apr}_{AT}(X)$,

(2) For any $a \in A$, $\underline{apr}_{A-\{a\}}(X) \neq \underline{apr}_{AT}(X)$,

then A is called the LA reduction of AT with respect to X, and is denoted as $Reduct(AT)_{L,X}$.

Definition 2.3 In an information table represented by Eq.(1), for any target concept $X \subseteq U$, if $A \subseteq AT$ satisfies the following two conditions:

(1) $\overline{apr}_{A}(X) = \overline{apr}_{AT}(X)$,

(2) For any $a \in A$, $\overline{apr}_{A-\{a\}}(X) \neq \overline{apr}_{AT}(X)$,

then A is called the UA reduction of AT with respect to X, and is denoted as $Reduct(AT)_{U,X}$.

3. Simplified Pawlak rough set (SPRS)

Over the past forty years, more and more RS models have been developed to address various data problems. However, most of these models are proposed using the same or similar rules. In this part, we come up with some new rules, and Pawlak RS can be simplified and equivalently redesigned as follows.

Theorem 3.1 In an information table represented by Eq.(1), $X \subseteq U$ is a target concept, then we have

$$\overline{apr}_{AT}(X) = \cup_{x\in X} [x]_{AT}, \quad (7)$$

$$= U - \{x \in X^C \mid [x]_{AT} \subseteq X^C\}. \quad (8)$$

Proof Firstly, let's prove that Eq. (7) is true.

($\Leftarrow$:) For each $x_0 \in \cup_{x\in X}[x]_{AT}$, there exists $x_1 \in X$ such that $x_0 \in [x_1]_{AT}$. Then we have $[x_1]_{AT} \cap X \neq \emptyset$ and $[x_1]_{AT} = [x_0]_{AT}$. So, it can be obtained that $[x_1]_{AT} \cap X \neq \emptyset$. Hence, $x_0 \in \{x \in U \mid [x]_{AT} \cap X \neq \emptyset\}$, i.e., $x_0 \in \overline{apr}_{AT}(X)$. Therefore, we have $\cup_{x\in X}[x]_{AT} \subseteq \overline{apr}_{AT}(X)$.

($\Rightarrow$:) For each $x_0 \in \{x \in U \mid [x]_{AT} \cap X \neq \emptyset\}$, we have $[x_0]_{AT} \cap X \neq \emptyset$. Then there exists $x_1 \in U$ such that $x_1 \in [x_0]_{AT}$ and $x_1 \in X$. Based on $x_1 \in [x_0]_{AT}$, one can find that $[x_1]_{AT} = [x_0]_{AT}$. Then we have $x_0 \in [x_1]_{AT}$ and $x_1 \in X$. Hence, $x_0 \in \cup_{x\in X}[x]_{AT}$, i.e., $\overline{apr}_{AT}(X) \subseteq \cup_{x\in X}[x]_{AT}$.

Secondly, let's prove that Eq. (8) holds.

($\Leftarrow$:) For each $x_0 \in U - \{x \in X^C \mid [x]_{AT} \subseteq X^C\}$, we have $x_0 \overline{\in} \{x \in X^C \mid [x]_{AT} \subseteq X^C\}$. Then $[x_0]_{AT} \cap X \neq \emptyset$. Hence, $x_0 \in \cup\{[x]_{AT} \in U/E_{AT} \mid [x]_{AT} \cap X \neq \emptyset\}$. By Eq. (3), it can be obtained that $x_0 \in \overline{apr}_{AT}(X)$, i.e., $U - \{x \in X^C \mid [x]_{AT} \subseteq X^C\} \subseteq \overline{apr}_{AT}(X)$.

($\Rightarrow$:) For each $x_0 \in \{x \in U \mid [x]_{AT} \cap X \neq \emptyset\}$, we have $[x_0]_{AT} \cap X \neq \emptyset$. Then $[x_0]_{AT} \nsubseteq X^C$ i.e., $x_0 \in U - \{x \in X^C \mid [x]_{AT} \subseteq X^C\}$. Therefore, $\overline{apr}_{AT}(X) \subseteq U - \{x \in X^C \mid [x]_{AT} \subseteq X^C\}$.

Theorem 3.2 In an information table represented by Eq.(1), $X \subseteq U$ is a target concept, then we have

$$\underline{apr}_{AT}(X) = \{x \in X \mid [x]_{AT} \subseteq X\}, \quad (9)$$

$$= X - \cup_{x\in X^C}[x]_{AT}. \quad (10)$$

Proof Similar to the proof of Theorem 3.1, it is immediate.

According to Theorems 3.1 and 3.2, TPRS can be simplified and equivalently defined based on new rules. That is, only those equivalence classes of the data in the target concept, rather than the data set, need to be calculated. Moreover, from Eq.s (7) and (10), the relationship between the equivalence class and the target concept does not need to be tested.

Theorem 3.3 In an information table represented by Eq.(1), $X \subseteq U$ is a target concept, then we have

$$Bou_{AT}(X) = \cup_{x\in\overline{apr}_{AT}(X)-X} [x]_{AT};$$

$$= \cup_{x\in\overline{apr}_{AT}(X^C)-X^C} [x]_{AT};$$

Corollary 3.1 In an information table represented by Eq.(1), $X \subseteq U$ is a target concept, then

$$Bou_{AT}(X) = Bou_{AT}(X^C).$$

4. Attribute reduction

Attribute reduction aims to remove redundant or unimportant attributes to reduce data dimensionality and complexity. This is of great significance for improving the efficiency of analyzing data and reducing over-fitting phenomena. In the theory of reduction, attribute reductions of UA and LA are very important and representative. So far, people have paid little attention to the relationship between them. In this part, we try to study the relationship between them and obtain the following interesting fact.

Theorem 4.1 In an information table represented by Eq.(1), $X \subseteq U$ is a target concept, then A is a LA reduction of AT with respect to X if and only if A is an UA reduction of AT with respect to X^C.

Proof ($\Rightarrow$:) Suppose that A is a LA reduction of AT with respect to X. Then we have

(1) $\underline{apr}_A(X) = \underline{apr}_{AT}(X)$,

(2) For any $a \in A$, $\underline{apr}_{A/\{a\}}(X) \neq \underline{apr}_{AT}(X)$.

Because $\underline{apr}_A(X) \cap \overline{apr}_A(X^C) = \emptyset$ and $\underline{apr}_A(X) = U - \overline{apr}_A(X^C)$, from equations $\underline{apr}_A(X) = \underline{apr}_{AT}(X)$ and $\underline{apr}_{A/\{a\}}(X) \neq \underline{apr}_{AT}(X)$, we can get $\overline{apr}_A(X^C) = \overline{apr}_{AT}(X^C)$ and $\overline{apr}_{A/\{a\}}(X^C) \neq \overline{apr}_{AT}(X^C)$. Then, we have

($1'$) $\overline{apr}_A(X^C) = \overline{apr}_{AT}(X^C)$,

($2'$) For any $a \in A$, $\overline{apr}_{A/\{a\}}(X^C) \neq \overline{apr}_{AT}(X^C)$.

Based on Definition 2.3, A is an UA reduction of AT with respect to X^C.

($\Leftarrow$:) Similarly, it is immediate.

In order to have a more intuitive understanding of Theorem 4.1, a specific example is employed as follows.

Example 4.1 Here is an information table, where $U = \{x_1, x_2, \cdots, x_8\}$, and $AT = \{a_1, a_2, a_3, a_4\}$. See Table 1 for details.

Table 1: An information table

OB	a_1	a_2	a_3	a_4
x_1	1	1	0	1
x_2	1	1	0	0
x_3	1	0	1	0
x_4	1	0	1	0
x_5	0	0	0	1
x_6	0	0	0	1
x_7	0	1	1	1
x_8	0	1	1	0

For $X = \{x_2, x_3, x_5, x_7\}$, based on Definition 2.2, one can find that $A = \{a_3, a_4\}$ is a LA reduction with respect to X. Meanwhile, according to Definition 2.3, it can be seen that $A = \{a_3, a_4\}$ is also the UA reduction with respect to X^C, i.e., $Reduct(AT)_{L,X} = Reduct(AT)_{U,X^C}$.

5. Algorithms

In the previous two sections, we discuss the equivalent characterization of TPRS, and prove that the attribute reductions of UA and LA are equivalent. Here, based on SPRS, we will design algorithms for calculating ULAs and reduction, and study the time complexity of these algorithms.

• Algorithm for calculating upper approximation (UA)

Here, two algorithms are designed to compute UA of Pawlak RS based on new rules. From Theorem 3.1, UA of SPRS can be computed by Eqs. (7) and (8). Based on Eqs. (7) and (8), algorithms 1 and 2 are respectively developed as follows.

Algorithm 1 An algorithm for computing $\overline{apr}_{AT}(X)$.

INPUT: An information table $I = (U, AT, \{V_a | a \in AT\}, \{f_a | a \in AT\})$, a target concept $X \subseteq U$;
OUTPUT: $\overline{apr}_{AT}(X)$.
  $\emptyset \leftarrow \overline{apr}_{AT}(X)$
  For $i = 1 : |X|; i <= |X|; i++$ do
    Computing $[x_i]_{AT}, x_i \in X$
    $\overline{apr}_{AT}(X) \leftarrow \overline{apr}_{AT}(X) \cup [x_i]_{AT}$

Algorithm 2 An algorithm for computing $\overline{apr}_{AT}(X)$.

Input: An information table $I = (U, AT, \{V_a|a \in AT\}, \{f_a|a \in AT\})$, a target concept $X \subseteq U$;
Output: $\overline{apr}_{AT}(X)$.
  $U \leftarrow \overline{apr}_{AT}(X)$
  For $i = 1 : |X^C|; i <= |X^C|; i++$ do
    Computing $[x_i]_{AT}, x_i \in X^C$
    $\overline{apr}_{AT}(X) \leftarrow \overline{apr}_{AT}(X) - [x_i]_{AT}$

• Algorithms for calculating lower approximation (LA)

Next, the algorithms are developed to calculate LA of Pawlak RS according to new rules. By Theorem 3.2, we provide two equivalent characterizations of LA of Pawlak RS. Based on Eqs. (9) and (10), algorithms 3 and 4 are developed for calculating LA of SPRS, respectively.

Algorithm 3 An algorithm for computing $\underline{apr}_{AT}(X)$.

Input: An information table $I = (U, AT, \{V_a|a \in AT\}, \{f_a|a \in AT\})$, a target concept $X \subseteq U$;
Output: $\underline{apr}_{AT}(X)$.
  $\emptyset \leftarrow \underline{apr}_{AT}(X)$
  For $i = 1 : |X|; i <= |X|; i++$ do
    Computing $[x_i]_{AT}, x_i \in X$
    If $[x_i]_{AT} \subseteq X$, then $\underline{apr}_{AT}(X) \leftarrow \underline{apr}_{AT}(X) \cup \{x_i\}$
    Otherwise, $i \leftarrow i + 1$

Algorithm 4 An algorithm for computing $\underline{apr}_{AT}(X)$.

Input: An information table $I = (U, AT, \{V_a|a \in AT\}, \{f_a|a \in AT\})$, a target concept $X \subseteq U$;
Output: $\underline{apr}_{AT}(X)$.
  $X \leftarrow \underline{apr}_{AT}(X)$
  For $i = 1 : |X^C|; i <= |X^C|; i++$ do
    Computing $[x_i]_{AT}, x_i \in X^C$
    $\underline{apr}_{AT}(X) \leftarrow \underline{apr}_{AT}(X) - [x_i]_{AT}$

• Algorithms for calculating attribute reduction

By Theorem 4.1, the LA reduction of the target set X is equal to the UA reduction of X^C, i.e., $Reduct(A)_{L,X} = Reduct(A)_{U,X^C}$. According to SPRS model, algorithms 5 and 6 are studied for computing $Reduct(A)_{L,X}$ and $Reduct(A)_{U,X^C}$, respectively.

Algorithm 5 An algorithm for computing $Reduct(AT)_{L,X}$.

Input: An information table $I = (U, AT, \{V_a|a \in AT\}, \{f_a|a \in AT\})$, and a target concept $X \subseteq U$;
Output: $Reduct(AT)_{L,X}$.
  $AT \leftarrow Reduct(AT)_{L,X}$
  For $i = 1 : |AT|; i <= |AT|; i++$ do
  If $\underline{apr}_{Reduct(AT)_{L,X}/\{a_i\}}(X) = \underline{apr}_{Reduct(AT)_{L,X}}(X)$, then $Reduct(AT)_{L,X} \leftarrow Reduct(AT)_{L,X} - \{a_i\}$
  Otherwise, $i \leftarrow i + 1$

Algorithm 6 An algorithm for computing $Reduct(AT)_{U,X^C}$.

INPUT: An information table $I = (U, AT, \{V_a|a \in AT\}, \{f_a|a \in AT\})$, and a target concept $X \subseteq U$;
OUTPUT: $Reduct(A)_{U,X^C}$.
$A \leftarrow Reduct(AT)_{U,X^C}$
For $i = 1 : |AT|; i <= |AT|; i++$ do
If $\overline{apr}_{Reduct(AT)_{U,X^C}/\{a_i\}}(X) = \overline{apr}_{Reduct(AT)_{U,X^C}}(X)$, then $Reduct(AT)_{U,X^C} \leftarrow Reduct(AT)_{U,X^C} - \{a_i\}$
Otherwise, $i \leftarrow i + 1$

Finally, we will study the complexity of all proposed algorithms. And the complexity of these algorithms is shown in Table 2, where the letter l represents the number of attributes in the information table.

Table 2: The time complexity of Algorithms

Algorithms	The time complexity
Algorithm 1	$O(\|X\|\|U\|)$
Algorithm 2	$O\left(\|X^C\|\|U\|\right) + O\left(\|X^C\|^2\right)$
Algorithm 3	$O(\|X\|\|U\|) + O\left(\|X\|^2\right)$
Algorithm 4	$O\left(\|X^C\|\|U\|\right)$
Algorithm 5	$O(l \times \|X\|\|U\|) + O\left(l \times \|X\|^2\right)$
Algorithm 6	$O\left(l \times \|X^C\|\|U\|\right)$

6. Experimental analysis

In this section, we choose nine data sets in UCI (http://archive.ics.uci.edu/ml/datasets.html) for experimental analysis. The details of data sets are listed in Table 3. All the experimental processes and results are completed by a private computer. And Table 4 shows the experimental operating environment including relevant parameters. To distinguish the time consumption, each of the nine data sets is divided into ten disjoint parts with equal size, and denoted as $U_1^{'}, U_2^{'}, \ldots, U_{10}^{'}$. Here, $U_i = \cup_{j=1}^{i} U_j^{'}$ $(i = 1, 2, \ldots, 10)$ are selected as ten universes with increasing sizes in sequence. And we randomly select 10%, 20%, $\cdots$, 90% of the data in each data set as target concepts with increasing scales and denote them as $X_1, X_2, \cdots, X_9$, respectively.

Table 3: The basic information of data sets

No.	Datasets	Objects	Attributes
1	Statlog	1000	24
2	OPPORTUNITY	2511	240
3	Seismic-bumps	2584	17
4	Page-blocks	5472	10
5	Thyroid Disease	7200	21
6	Mushroom	8124	23
7	Occupancy-Estimation	10129	17
8	Magic	19020	10
9	Shuttle	57999	9

Table 4: Specific information about the operating environment

Name	Model	Parameter
CPU	Intel(R) Core(TM) i5-6300HQ	2.30 GHz
Platform	Python	3.11
System	Windows10	64 bit
Memory	SAMSUNG DDR4	8 GB; 2666 MHz
Hard Disk	SAMSUNG SSD	256 GB

6.1. Experimental analysis on SPRS

• Experimental analysis on UA

Table 5: The time consumption for computing UA of SPRS by Eq.s (7)-(8).

No.s	TC	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
1	7	0.0090	0.0199	0.0309	0.0389	0.0499	0.0598	0.0728	0.0748	0.0997
	8	0.1027	0.0888	0.0778	0.0658	0.0529	0.0439	0.0299	0.0189	0.0091
2	7	0.0668	0.1296	0.2014	0.2703	0.3471	0.4169	0.5046	0.5696	0.6483
	8	0.7071	0.6264	0.5585	0.4612	0.3790	0.3156	0.2284	0.1417	0.0708
3	7	0.0658	0.1336	0.2035	0.2912	0.3326	0.4374	0.5346	0.5768	0.6565
	8	0.7640	0.6570	0.5814	0.4932	0.4079	0.3010	0.2274	0.1496	0.0718
4	7	0.2484	0.5116	0.7486	0.9993	1.6805	1.9774	2.0491	2.5990	2.6419
	8	2.6895	2.2616	1.9689	2.1535	1.7150	1.2965	0.9689	0.5934	0.3870
5	7	0.5417	1.3880	1.7297	1.8127	2.3061	2.7377	3.1936	3.7390	4.1542
	8	7.5896	5.1389	3.8664	3.0193	2.4611	2.0167	1.4372	0.9515	0.4638
6	7	0.5160	1.0454	1.6720	2.1504	2.6990	3.2607	3.8388	4.4334	5.0225
	8	5.7560	5.1187	4.4540	3.6256	3.0050	2.3927	1.7656	1.1779	0.5799
7	7	0.9032	1.8487	2.8131	3.7634	4.7945	5.9183	6.9217	8.1321	12.9093
	8	10.6195	10.7996	8.2244	7.0329	5.8968	4.2626	3.1816	2.1081	1.0752
8	7	3.3971	7.4120	10.5654	14.1877	18.2195	22.3109	26.8448	30.9229	36.3427
	8	39.6670	35.2671	30.2703	26.0702	21.3641	17.0093	12.8212	7.8749	3.8194
9	7	31.7707	63.9949	95.3883	127.1779	158.7870	188.8332	224.6585	255.5586	286.5385
	8	294.7623	260.0291	225.4803	185.2510	161.9970	118.9991	99.2265	66.7298	35.4209

According to Theorem 3.1, UA of SPRS can be calculated by Eq.s (7)-(8), respectively. From Table 5, as the size of target concept increases, the time consumption for computing UA by Eq. (7) (TC 7) continues to increase, while the time consumption related to Eq. (8) (TC 8) gradually decreases. Moreover, when the scale of target concept does not exceed about half of the size of data set, the TC 7 is shorter than TC 8. Otherwise, the TC 7 is longer than TC 8.

• Experimental analysis on LA

Table 6: The time consumption for computing LA of SPRS by Eq.s (9)-(10).

No.s	TC	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
1	9	0.0130	0.0260	0.0400	0.0570	0.0730	0.0900	0.1060	0.1250	0.1430
	10	0.1300	0.1170	0.1036	0.0882	0.0730	0.0590	0.0460	0.0299	0.0150
2	9	0.0940	0.1966	0.3065	0.4200	0.5410	0.6578	0.7794	0.9018	1.0248
	10	0.9337	0.8319	0.7393	0.6364	0.5356	0.4255	0.3215	0.2130	0.1070
3	9	0.0598	0.1237	0.2254	0.2443	0.3271	0.3560	0.4528	0.5535	0.6054
	10	0.6054	0.5635	0.4747	0.3969	0.3101	0.2314	0.1945	0.1267	0.0629
4	9	0.3819	0.7745	1.2073	1.6392	2.0800	2.5601	2.9845	3.4683	4.2163
	10	3.7238	3.3226	2.8803	2.5276	2.0698	1.6414	1.2713	0.8206	0.4139
5	9	0.6368	0.8477	1.2766	1.7662	2.2872	2.7233	3.3223	3.8874	4.4353
	10	5.3591	3.6743	3.2328	2.7668	2.2875	1.8282	1.3843	0.9221	0.4478
6	9	0.6104	1.6915	2.2370	2.6958	3.4318	5.3048	6.0937	6.2573	7.0032
	10	6.9305	6.9335	5.5166	4.1609	3.4767	3.1655	2.3198	1.5418	0.6742
7	9	1.2726	2.9893	4.4369	7.3680	9.1030	11.0936	13.6775	12.5804	14.8322
	10	13.4269	13.2416	12.2899	10.6051	8.8318	7.9067	4.2511	2.8504	1.7124
8	9	4.0318	8.3645	13.1352	18.1427	22.7474	28.1915	33.6920	39.0301	44.1038
	10	40.7255	35.7799	31.4716	26.8861	22.2527	17.8046	13.2332	8.7302	4.3070
9	9	31.6491	64.6309	98.9135	154.3552	196.4386	239.5567	285.4801	327.9593	376.5661
	10	333.0028	293.8888	254.0872	213.9053	176.4765	139.8453	102.9273	68.0112	34.0175

According to Theorem 3.2, LA of SPRS can be calculated by Eq.s (9)-(10), respectively. From Table 6, as the size of target concept increases, the time consumption related to Eq. (9) (TC 9) continues to increase, while the time consumption related to Eq. (10) (TC 10) gradually decreases. Moreover, when the scale of target concept does not exceed about half of the size of data set, the TC 9 is shorter than TC 10. Otherwise, the TC 9 is longer than TC 10.

• Experimental analysis on attribute reduction

Table 7: The time consumption for computing $Reduct(AT)_{L,X}$ and $Reduct(AT)_{U,X^C}$

No.s	TC	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
1	$R_{L,X}$	3.2281	3.5300	3.8876	4.3105	4.7614	5.1209	5.5486	5.9723	6.3550
	R_{U,X^C}	6.0903	5.7627	5.4615	5.1000	4.7226	4.3464	4.0012	3.6548	3.2454
2	$R_{L,X}$	206.5144	232.6477	259.0989	286.7574	317.2264	345.0704	373.5206	403.5474	434.0623
	R_{U,X^C}	414.1914	385.8488	361.9456	335.2094	312.5853	285.2354	259.5804	235.0160	208.0597
3	$R_{L,X}$	3.7879	5.4664	6.9145	7.7742	9.4039	11.5990	12.0388	13.1311	15.6172
	R_{U,X^C}	16.2794	12.7419	10.5199	10.8819	8.6798	8.4374	6.7260	5.9700	3.9415
4	$R_{L,X}$	13.2145	18.4506	23.9634	29.3443	34.8807	40.5286	46.6336	52.6181	61.1621
	R_{U,X^C}	54.1312	49.1370	44.1046	39.5836	34.6065	29.4209	24.1639	18.5800	13.5845
5	$R_{L,X}$	33.8745	51.7991	69.4297	84.9461	107.9610	130.3369	152.9633	179.9613	199.6641
	R_{U,X^C}	183.7521	164.5374	147.4273	125.9613	104.9916	88.9134	61.9469	45.9164	29.4432
6	$R_{L,X}$	19.6500	31.1298	39.9931	50.7323	61.6621	73.4616	89.0507	98.2216	111.3283
	R_{U,X^C}	101.8079	96.5438	82.6852	73.2971	62.4609	52.3779	41.2986	30.6839	20.5791
7	$R_{L,X}$	37.1180	60.2795	81.6254	93.1651	124.1684	159.1916	186.1652	221.9416	263.4631
	R_{U,X^C}	224.8226	198.5517	175.6727	155.4913	137.4981	103.1916	79.9411	54.9463	31.9493
8	$R_{L,X}$	71.7859	127.0945	186.7164	246.0780	307.0746	371.3767	441.3410	509.3631	579.4085
	R_{U,X^C}	520.4219	461.0685	404.9159	350.6180	294.2670	239.1114	185.3126	129.6961	75.8641
9	$R_{L,X}$	476.8629	917.4517	1482.0596	2046.3941	2469.1228	2930.8462	3377.9134	3801.9231	4394.9441
	R_{U,X^C}	4677.5042	4086.8970	3533.2387	3091.9463	2631.6583	2049.1631	1599.6134	996.4961	493.4955

Now, let's analyze the time consumption of attribute reduction. By Theorem 4.1, $Reduct(AT)_{L,X} = Reduct(AT)_{U,X^C}$. In what follows, we will compare the time consumption with respect to $Reduct(AT)_{L,X}$ and $Reduct(AT)_{U,X^C}$. From Table 7, as the size of target concept increases, the time consumption related to $Reduct(AT)_{L,X}$ (TC $R_{L,X}$) continues to increase, while the time consumption related to $Reduct(AT)_{U,X^C}$ (TC R_{U,X^C}) gradually decreases. Moreover, when the scale of the target concept exceeds about half of that of data set, TC R_{U,X^C} is shorter than TC $R_{L,X}$. Otherwise, TC R_{U,X^C} is longer than TC $R_{L,X}$.

6.2. Comparative analysis between SPRS and traditional RSs

TPRS, as we all know, is good at analyzing complete symbolic data. When the information table contains certain errors or important information is missing, TPRS can not make effective analysis. For this reason, variable precision RS (VPRS) is introduced [27]. As an important extension of TPRS, VPRS can effectively handle data with noise.

We know that TPRS has very strict requirements for knowledge classification and lacks fault tolerance. In order to address this issue, probabilistic RS (PRS) is proposed [28]. The classification of PRS is not entirely correct or uncertain, but it has a certain degree of error tolerance.

TPRS divides objects in the universe into known concepts through a single granular structure or an equivalent relationship, and then the unknown concepts are represented by using the known concepts. However, there are often multiple granular structures in a data set, so TPRS needs to be promoted, and multi-granulation RS (MGRS) based on multiple granular structures is developed and widely studied [13].

For TPRS model, all objects in the data set participate in the calculation of ULAs, which leads to low efficiency in analyzing the data. Y.H. Qian, et al. initially propose the local RS (LRS) [29], in which only the data in the target concept, rather than all data in the data set, is focused on. This helps greatly improve the computational efficiency of RS model.

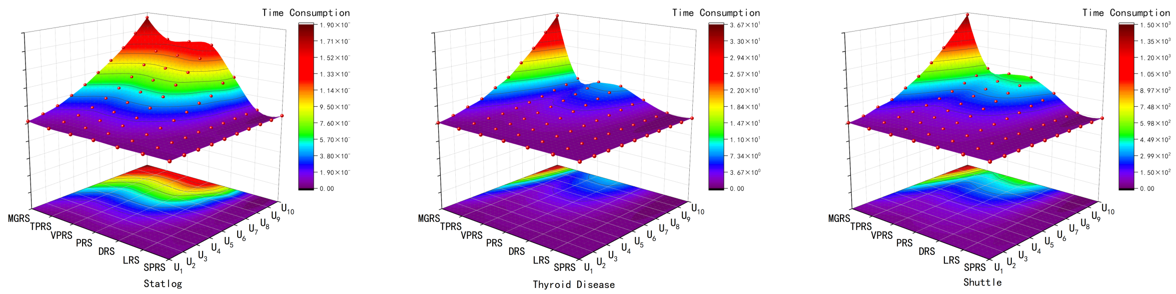
Traditional data mining methods often encounter difficulties in balancing the efficiency and accuracy of data processing. To tackle this challenge, Q.Z. Kong et al. propose a novel data processing technique called the DMF strategy, and further introduce a DMF-based RS (DRS) model [30]. The DRS model enables rapid and precise data analysis.

The six types of RS models mentioned above have been widely discussed and are very representative. Here, from the perspective of the computational efficiency, SPRS will be compared and studied with these six types of RSs.

• Comparative analysis on UA

Table 8: The time consumption for computing UA in different universes

No.s	U	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS	U	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS
	U_1	0.0049	0.0010	0.0010	0.0020	**0.0007**	0.0010	0.0010	U_6	0.0770	0.0490	0.0500	0.0500	0.0265	**0.0040**	**0.0040**
	U_2	0.0120	0.0051	0.0045	0.0050	0.0025	**0.0020**	**0.0020**	U_7	0.0991	0.0680	0.0700	0.0690	0.0381	**0.0040**	0.0046
1	U_3	0.0219	0.0110	0.0110	0.0120	0.0057	**0.0020**	**0.0020**	U_8	0.1240	0.0900	0.0910	0.0909	0.0461	**0.0050**	**0.0050**
	U_4	0.0376	0.0201	0.0210	0.0199	0.0100	**0.0019**	0.0020	U_9	0.1540	0.1170	0.1150	0.1165	0.0683	0.0060	**0.0050**
	U_5	0.0550	0.0330	0.0340	0.0330	0.0170	0.0040	**0.0030**	U_{10}	0.1879	0.1405	0.1426	0.1420	0.0743	0.0070	**0.0060**
	U_1	0.0250	0.0080	0.0080	0.0080	0.0045	**0.0040**	**0.0040**	U_6	0.5098	0.3801	0.3734	0.3739	0.1813	0.0290	**0.0280**
	U_2	0.0810	0.0360	0.0380	0.0370	0.0173	0.0090	**0.0080**	U_7	0.6726	0.5129	0.5081	0.5089	0.2502	**0.0330**	**0.0330**
2	U_3	0.1584	0.0900	0.0920	0.0915	0.0403	0.0140	**0.0130**	U_8	0.8550	0.6629	0.6584	0.6570	0.3328	**0.0370**	0.0375
	U_4	0.2509	0.1679	0.1676	0.1661	0.0893	**0.0190**	**0.0190**	U_9	1.0465	0.8189	0.8279	0.8194	0.4132	**0.0420**	**0.0420**
	U_5	0.3584	0.2621	0.2619	0.2625	0.1476	0.0241	**0.0230**	U_{10}	1.2422	0.9959	1.0035	0.9919	0.4834	**0.0465**	0.0480
	U_1	0.0302	0.0050	0.0050	0.0050	0.0030	**0.0020**	**0.0020**	U_6	0.3112	0.1963	0.1926	0.1896	0.0978	0.0151	**0.0140**
	U_2	0.0499	0.0219	0.0209	0.0219	0.0102	0.0050	**0.0040**	U_7	0.3880	0.2594	0.2663	0.2563	0.1193	**0.0170**	**0.0170**
3	U_3	0.0958	0.0459	0.0473	0.0479	0.0287	0.0080	**0.0071**	U_8	0.4992	0.3505	0.3385	0.3301	0.1721	**0.0190**	0.0219
	U_4	0.1561	0.0828	0.0840	0.0859	0.0462	0.0101	**0.0092**	U_9	0.5966	0.4260	0.4220	0.4169	0.2803	0.0219	**0.0209**
	U_5	0.2176	0.1410	0.1388	0.1366	0.0794	0.0121	**0.0120**	U_{10}	0.7081	0.5339	0.5302	0.5097	0.3102	0.0251	**0.0239**
	U_1	0.1150	0.0410	0.0412	0.0407	0.0183	0.0200	**0.0170**	U_6	1.9470	1.3870	1.3723	1.3757	0.6284	0.1020	**0.1009**
	U_2	0.3229	0.1669	0.1665	0.1660	0.0815	0.0380	**0.0369**	U_7	2.5303	1.8849	1.8628	1.8292	0.9313	**0.1170**	0.1220
4	U_3	0.6244	0.3667	0.3656	0.3645	0.1832	0.0549	**0.0530**	U_8	3.2035	2.4287	2.3960	2.3846	1.2937	0.1360	**0.1311**
	U_4	0.9664	0.6284	0.6166	0.6160	0.3144	0.0710	**0.0680**	U_9	3.9808	3.0246	2.9997	3.0129	1.6004	0.1500	**0.1471**
	U_5	1.4498	0.9914	0.9785	0.9823	0.4896	0.0879	**0.0860**	U_{10}	5.0190	3.7553	3.7307	3.7035	1.8269	**0.1629**	0.1639
	U_1	0.2786	0.0600	0.0610	0.0620	0.0321	0.0300	**0.0250**	U_6	12.2507	2.7163	2.8290	2.9292	1.3346	0.0327	**0.0279**
	U_2	1.5709	0.3043	0.3877	0.3819	0.1612	0.0301	**0.0247**	U_7	17.4931	3.8275	3.9450	4.0489	1.8659	0.0334	**0.0286**
5	U_3	3.5782	0.7661	0.6369	0.6391	0.3771	0.0298	**0.0257**	U_8	23.0251	5.1581	5.2554	5.2603	2.4264	0.0335	**0.0300**
	U_4	5.5122	1.0001	1.0552	1.1778	0.5013	0.0316	**0.0263**	U_9	29.3175	6.2707	6.4650	6.4835	3.2331	0.0351	**0.0296**
	U_5	8.9810	1.8186	1.9186	1.8351	0.9616	0.0327	**0.0276**	U_{10}	36.1035	7.8360	7.9989	8.0053	3.6435	0.0347	**0.0303**
	U_1	0.4119	0.0888	0.0947	0.0768	0.0435	0.0369	**0.0339**	U_6	8.5212	2.5851	2.5761	2.5951	1.2106	**0.1935**	0.2014
	U_2	1.8800	0.3760	0.3201	0.4757	0.1731	**0.0708**	0.0778	U_7	11.3865	3.5714	3.5216	3.5415	1.7320	0.2214	**0.2184**
6	U_3	3.3411	0.6772	0.6712	0.6892	0.3332	**0.0977**	0.0987	U_8	14.0334	4.6316	4.5429	4.5977	2.1153	**0.2603**	0.2892
	U_4	5.0346	1.1719	1.1619	1.2128	0.6106	0.1316	**0.1306**	U_9	16.8270	5.8214	5.7835	5.7606	2.8113	0.2932	**0.2833**
	U_5	6.7070	1.8500	1.8331	1.8610	0.9348	**0.1626**	0.1726	U_{10}	20.2518	7.1499	7.1419	7.0641	3.4301	0.3221	**0.3141**
	U_1	0.2972	0.1139	0.1118	0.1117	0.0503	0.0529	**0.0459**	U_6	9.4652	3.1655	3.1641	3.1946	1.6240	0.0821	**0.0633**
	U_2	1.1941	0.3491	0.3164	0.3116	0.1691	0.0649	**0.0503**	U_7	13.6133	4.1694	4.1946	4.1550	2.0133	0.0860	**0.0684**
7	U_3	2.5913	0.8496	0.8913	0.8794	0.4182	0.0694	**0.0543**	U_8	15.9950	5.6943	5.6114	5.6344	2.7217	0.0896	**0.0731**
	U_4	4.9613	1.4961	1.4891	1.4613	0.7015	0.0764	**0.0561**	U_9	18.9160	7.6131	7.8123	7.7994	3.6532	0.0934	**0.0811**
	U_5	7.6616	2.3691	2.4316	2.3112	1.1823	0.0799	**0.0598**	U_{10}	22.2164	9.3465	9.2613	9.3946	4.1694	0.1005	**0.0901**
	U_1	1.1481	0.4098	0.4155	0.4129	0.1913	0.1925	**0.1866**	U_6	19.2992	14.3993	13.7654	13.9201	7.8642	1.0942	**1.0500**
	U_2	3.1771	1.5618	1.5779	1.5644	0.8133	0.3762	**0.3642**	U_7	25.3074	18.9473	18.9159	18.9259	9.6641	**1.2554**	1.2594
8	U_3	5.9791	3.3784	3.3794	3.6787	1.6217	0.5490	**0.5238**	U_8	32.4064	24.5154	24.7594	24.6844	12.9143	1.5232	**1.4393**
	U_4	9.7787	6.2244	6.0717	6.0702	3.2132	0.7220	**0.7063**	U_9	40.4706	32.7425	31.8664	32.3235	16.0019	1.6790	**1.6452**
	U_5	14.4554	9.5321	9.5351	9.7534	4.7648	0.9175	**0.8798**	U_{10}	49.3040	39.9544	40.6798	40.1505	20.8746	1.8429	**1.8382**
	U_1	11.0993	3.9320	3.6834	3.6783	1.8065	1.7979	**1.6558**	U_6	507.7071	178.0702	166.2304	164.1308	85.9466	2.1563	**1.6566**
	U_2	53.2522	29.0141	15.8673	24.6637	15.4168	1.8200	**1.6574**	U_7	700.6612	244.7367	239.6338	231.2859	118.6937	2.1447	**1.6866**
9	U_3	142.3747	50.4673	47.6812	40.3657	24.2210	1.8531	**1.6509**	U_8	931.4923	339.6167	304.4991	312.5235	167.9433	2.2124	**1.7551**
	U_4	206.6796	72.2806	64.6249	66.3744	38.3385	1.9314	**1.6792**	U_9	1190.5097	421.6650	388.0410	387.2271	204.6150	2.2606	**1.7206**
	U_5	354.5460	120.9333	112.4814	105.2692	63.1389	2.0435	**1.6780**	U_{10}	1462.7969	505.6805	474.1353	482.3979	267.7736	2.3278	**1.7996**



(a) (b) (c)

Figure 1: The time consumption for computing UA in different universes.

Now, let's analyze the impact of data size on the time spent calculating UA. Figure 1 can be obtained by Table 8.

From Figure 1 and Table 8, we can observe the following phenomenas.

(1) When calculating the UA of Pawlak RS, the computational efficiency of SPRS is much higher than that of TPRS. And as the size of universe increases, the advantage of SPRS in terms of computational efficiency become increasingly apparent. For example, in the first data set, if U_{10} is taken as the universe, the time consumption of SPRS is only 1/23 of that of TPRS. In the ninth data set, when considering U_{10} as the universe, the computing efficiency of SPRS model is 280 times that of TPRS.

(2) Regardless of the size of the universe, calculating the UA of SPRS always takes less time than that of MGRS, VPRS, PRS, and DRS. For the SPRS and LRS models, regardless of whether the size of universe is large or small, in most cases, SPRS is more effective in analyzing data than LRS.

Table 9: The time consumption for computing UA in different target concepts

No.s	X	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS	X	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS
	X_1	0.2190	0.1519	0.1545	0.1540	0.0802	0.0130	**0.0125**	X_6	0.5421	0.1629	0.1649	0.1641	0.0872	0.0959	**0.0529**
	X_2	0.2835	0.1536	0.1562	0.1560	0.0811	0.0280	**0.0270**	X_7	0.6045	0.1640	0.1661	0.1649	0.0885	0.1150	**0.0439**
1	X_3	0.3496	0.1550	0.1582	0.1589	0.0848	0.0440	**0.0439**	X_8	0.6620	0.1653	0.1709	0.1689	0.0893	0.1340	**0.0299**
	X_4	0.4185	0.1605	0.1629	0.1630	0.0850	0.0610	**0.0570**	X_9	0.7024	0.1679	0.1706	0.1655	0.0910	0.1490	**0.0189**
	X_5	0.4850	0.1619	0.1649	0.1635	0.0861	0.0786	**0.0720**								
	X_1	1.4368	1.0338	1.0189	1.0190	0.5105	0.0939	**0.0929**	X_6	3.2466	1.0845	1.0968	1.0883	0.5312	0.6417	**0.3790**
	X_2	1.8116	1.0428	1.0486	1.0449	0.5184	0.1999	**0.1923**	X_7	3.6496	1.0896	1.1179	1.1186	0.5294	0.7571	**0.3156**
2	X_3	2.2238	1.0528	1.0595	1.0554	0.5213	0.3075	**0.2939**	X_8	4.0151	1.0976	1.1013	1.1017	0.5241	0.8870	**0.2284**
	X_4	2.6179	1.0598	1.0679	1.0699	0.5225	0.4148	**0.3974**	X_9	4.3131	1.1228	1.1388	1.1330	0.5388	1.0140	**0.1417**
	X_5	2.9562	1.0894	1.1022	1.1026	0.5274	0.5248	**0.4888**								
	X_1	0.8559	0.5416	0.5534	0.5326	0.2684	**0.0511**	0.0531	X_6	2.0562	0.6025	0.5966	0.6003	0.2794	0.3626	**0.2231**
	X_2	1.1780	0.5615	0.5591	0.5545	0.2735	0.1097	**0.1009**	X_7	2.3434	0.6031	0.6107	0.5974	0.2736	0.4121	**0.1874**
3	X_3	1.3364	0.5527	0.5663	0.5575	0.2717	0.1614	**0.1528**	X_8	2.5706	0.6164	0.6197	0.6543	0.2801	0.4983	**0.1296**
	X_4	1.6257	0.5595	0.5771	0.5744	0.2798	0.2154	**0.2064**	X_9	2.7953	0.6313	0.6212	0.6124	0.2818	0.5510	**0.0718**
	X_5	1.8690	0.5905	0.5867	0.5836	0.2742	0.2894	**0.2543**								
	X_1	5.8792	3.9184	3.9150	3.9308	1.7901	0.3745	**0.3599**	X_6	12.9475	4.2426	4.1931	4.2610	2.0311	2.4893	**1.7150**
	X_2	7.1308	4.0589	4.0237	4.0311	1.8053	0.7730	**0.7555**	X_7	14.4307	4.2722	4.2547	4.2610	2.0374	2.9795	**1.2965**
4	X_3	8.5756	4.0830	4.1097	4.1589	1.8732	1.1765	**1.1254**	X_8	15.5098	4.3396	4.3514	4.3490	2.1391	3.4743	**0.9689**
	X_4	10.1202	4.1113	4.1213	4.1734	1.9346	1.6080	**1.5308**	X_9	16.8898	4.5493	4.5380	4.5328	2.0413	4.0861	**0.5934**
	X_5	11.6150	4.1820	4.1949	4.2004	2.0235	2.0695	**1.9548**								
	X_1	29.3222	5.5314	5.5237	5.5177	2.8105	0.5290	**0.5243**	X_6	32.2922	6.2811	6.2094	6.1037	2.9174	3.1774	**2.0167**
	X_2	29.7587	5.8425	5.5441	5.5236	2.8163	1.0573	**1.0489**	X_7	32.2420	6.2127	6.4682	5.9901	2.9590	3.7122	**1.4372**
5	X_3	30.2514	5.7976	5.6540	5.7032	2.8291	1.5943	**1.5788**	X_8	32.1920	6.5345	6.7246	5.9503	2.8313	4.2409	**0.9515**
	X_4	31.4110	6.0991	5.9335	6.0396	2.8312	2.1180	**2.0924**	X_9	33.9809	6.7528	6.9122	6.2283	2.9385	4.7672	**0.4638**
	X_5	31.0091	6.3089	6.1935	6.1457	2.8646	2.6527	**2.6198**								
	X_1	21.4891	8.4300	8.2789	8.3985	4.7103	1.0334	**0.7650**	X_6	36.9082	10.1321	10.7052	8.8166	5.1191	5.2819	**3.0050**
	X_2	23.5530	12.8912	7.8919	7.2885	5.3236	**1.2816**	1.7124	X_7	37.7108	9.3622	9.8598	9.3485	5.3015	6.4037	**2.3927**
6	X_3	23.1451	6.7808	6.7559	6.8367	5.3325	2.3507	**1.9488**	X_8	44.6270	11.9871	9.4324	9.6242	5.1039	7.5806	**1.7656**
	X_4	27.9014	8.2091	10.8613	8.3506	5.6181	4.6526	**3.1256**	X_9	51.0729	9.4170	9.3907	9.4876	5.0014	12.1351	**1.1779**
	X_5	38.4137	12.0418	8.9448	10.8121	5.1380	4.8243	**3.6034**								
	X_1	15.5571	11.4820	11.2153	11.0492	5.1362	**0.9994**	1.0031	X_6	16.2100	12.1670	11.4517	11.3014	5.2349	6.0975	**4.2626**
	X_2	15.7440	11.3704	11.2327	11.0889	5.1391	**2.1382**	2.2145	X_7	16.4182	12.3213	11.6579	11.7709	5.2147	6.9636	**3.1816**
7	X_3	16.4055	11.3409	11.6087	10.8678	5.3152	3.0852	**3.0070**	X_8	17.3124	12.5252	12.0586	12.1253	5.2310	8.1085	**2.1081**
	X_4	16.3423	11.9241	11.3843	10.8805	5.1664	4.1724	**3.9634**	X_9	17.3570	12.4472	12.2906	12.5973	5.2734	9.1771	**1.0752**
	X_5	16.4115	12.0695	11.3687	11.2634	5.2135	**5.2014**	5.2337								
	X_1	56.3389	43.6799	42.9756	42.9810	22.6234	4.1339	**4.0143**	X_6	131.2026	47.3810	47.8464	47.6750	23.9642	27.9608	**17.0093**
	X_2	71.2648	43.3129	43.3171	43.3921	22.2294	8.3645	**8.1209**	X_7	145.2854	47.9516	47.8703	47.8510	24.4638	33.1917	**12.8212**
8	X_3	86.6800	45.8092	46.0380	45.8538	23.5320	13.2110	**12.4836**	X_8	156.5767	48.2574	48.4319	48.4312	24.5546	38.6455	**7.8749**
	X_4	101.6271	46.2532	46.4191	46.3489	23.1056	18.1639	**16.6935**	X_9	169.8781	48.9584	49.1646	49.2505	24.5188	44.1797	**3.8194**
	X_5	116.5740	46.9581	47.3378	46.8932	23.1946	23.1492	**21.3188**								
	X_1	572.8750	344.4470	343.0540	338.9970	169.1543	32.1548	**31.7707**	X_6	673.0085	364.1453	366.8894	367.1577	178.9563	192.7906	**118.9991**
	X_2	570.1477	350.3151	356.3165	343.3970	171.2638	64.0103	**63.9949**	X_7	711.1030	358.1425	370.6119	378.1878	179.6374	225.3054	**99.2265**
9	X_3	603.1796	367.6490	357.3666	339.1443	172.1134	96.6496	**95.3883**	X_8	744.4520	365.4627	391.6371	378.8494	180.2652	257.1951	**66.7298**
	X_4	629.4101	366.3897	354.2629	359.3774	173.1349	129.1476	**127.1779**	X_9	731.1933	361.8157	393.9296	371.8013	181.2564	289.2644	**35.4209**
	X_5	663.2179	368.8275	353.2547	370.7721	177.1946	160.6731	**158.7870**								

Next, let's analyze the impact of the size of target concept on the time spent calculating UA. According to Table 5, if the scale of the target concept is smaller, Algorithm 1 is chosen to analyze the data. Otherwise, Algorithm 2 is used to process the data. From Figure 2 and Table 9, we can get the following conclusions.

(1) The time taken to obtain the UA by using SPRS is always shorter than that by using TPRS, which is not affected by the size of the target concept. And the less data the target concept contains, the greater the advantage of the SPRS model is. For example, if the target concept is X_1, the efficiency of SPRS often exceeds that of TPRS by more than ten times.

(2) Regardless of the size of target concept, calculating the UA of SPRS always takes less time than that of MGRS, VPRS, PRS, and DRS. Additionally, in the vast majority of cases, SPRS analyzes data faster than LRS.

- Comparative analysis on LA

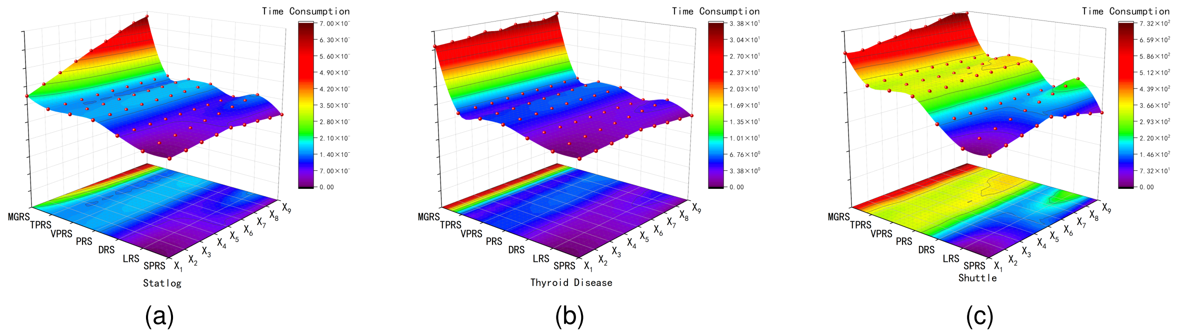


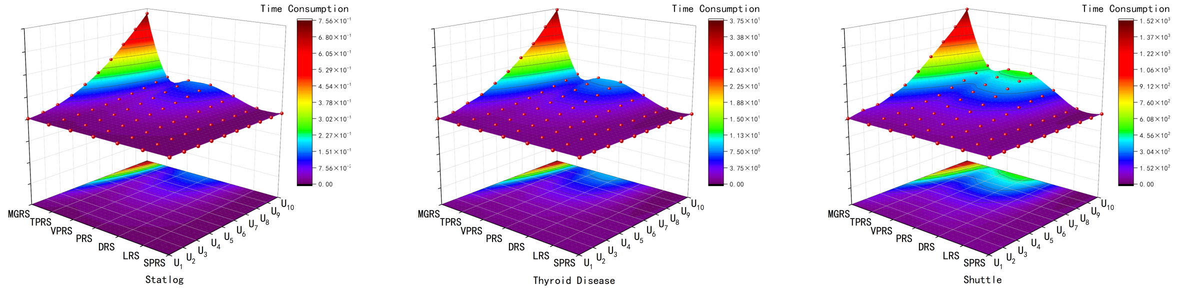
Figure 2: The time consumption for computing UA in different target concepts

Table 10: The time consumption for computing LA in different universes

No.s	U	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS	U	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS
1	U_1	0.0050	0.0010	0.0010	0.0010	**0.0005**	0.0010	0.0010	U_6	0.2665	0.0480	0.0500	0.0500	0.0243	**0.0030**	0.0040
	U_2	0.0240	0.0039	0.0050	0.0050	0.0021	**0.0010**	**0.0010**	U_7	0.3659	0.0680	0.0689	0.0690	0.0343	0.0050	**0.0040**
	U_3	0.0590	0.0110	0.0110	0.0111	0.0057	**0.0020**	**0.0020**	U_8	0.4890	0.0890	0.0910	0.0915	0.0427	**0.0050**	**0.0050**
	U_4	0.1120	0.0199	0.0211	0.0210	0.0102	0.0030	**0.0020**	U_9	0.6148	0.1160	0.1160	0.1160	0.0543	**0.0060**	**0.0060**
	U_5	0.1826	0.0319	0.0340	0.0340	0.0171	**0.0030**	**0.0030**	U_{10}	0.7518	0.1395	0.1443	0.1420	0.0774	**0.0060**	0.0070
2	U_1	0.0260	0.0080	0.0080	0.0090	0.0050	**0.0040**	**0.0040**	U_6	1.6108	0.3761	0.3749	0.3721	0.1984	**0.0280**	**0.0280**
	U_2	0.1556	0.0360	0.0370	0.0379	0.0172	**0.0090**	**0.0090**	U_7	2.2093	0.5128	0.5088	0.5065	0.2513	**0.0330**	**0.0330**
	U_3	0.3829	0.0890	0.0920	0.0910	0.0430	**0.0140**	**0.0140**	U_8	2.9138	0.6548	0.6596	0.6568	0.3135	**0.0370**	**0.0370**
	U_4	0.7013	0.1639	0.1677	0.1656	0.0812	0.0189	**0.0180**	U_9	3.7019	0.8205	0.8213	0.8278	0.4002	0.0420	**0.0417**
	U_5	1.1064	0.2629	0.2605	0.2574	0.1402	**0.0239**	0.0240	U_{10}	4.5288	0.9898	1.0048	0.9900	0.4950	0.0470	**0.0450**
3	U_1	0.0180	0.0050	0.0040	0.0050	0.0030	**0.0020**	0.0030	U_6	1.0165	0.1846	0.1876	0.1876	0.0947	**0.0160**	0.0172
	U_2	0.1077	0.0199	0.0229	0.0210	0.0057	0.0071	**0.0050**	U_7	1.3892	0.2594	0.2593	0.2543	0.1335	0.0180	**0.0170**
	U_3	0.2374	0.0476	0.0481	0.0550	0.0210	0.0081	**0.0070**	U_8	1.8383	0.3439	0.3407	0.3281	0.1785	**0.0199**	**0.0199**
	U_4	0.4318	0.0846	0.0952	0.0798	0.0448	0.0110	**0.0100**	U_9	2.3718	0.4221	0.4109	0.4213	0.2294	**0.0219**	**0.0219**
	U_5	0.6899	0.1318	0.1366	0.1355	0.0620	0.0130	**0.0122**	U_{10}	2.9907	0.5128	0.5147	0.5136	0.2613	0.0249	**0.0235**
4	U_1	0.1210	0.0406	0.0420	0.0411	0.0220	0.0190	**0.0185**	U_6	6.2284	1.3585	1.3809	1.3715	0.7024	**0.1010**	**0.1010**
	U_2	0.6325	0.1650	0.1681	0.1659	0.0835	**0.0379**	0.0380	U_7	8.5080	1.8535	1.8419	1.8319	0.9164	**0.1170**	0.1230
	U_3	1.4995	0.3619	0.3659	0.3640	0.1923	**0.0550**	**0.0550**	U_8	11.1829	2.3813	2.3863	2.3835	1.1438	0.1329	**0.1320**
	U_4	2.7089	0.6149	0.6184	0.6160	0.3152	0.0700	**0.0680**	U_9	14.1754	2.9705	3.0201	3.0029	1.5130	**0.1470**	**0.1470**
	U_5	4.4209	0.9915	0.9775	0.9734	0.4729	**0.0870**	**0.0870**	U_{10}	17.5238	3.7074	3.7334	3.6828	1.8842	**0.1636**	0.1651
5	U_1	0.2808	0.0600	0.0620	0.0606	0.0311	0.0309	**0.0300**	U_6	13.0299	2.7716	2.8590	2.7395	1.3346	0.0341	**0.0326**
	U_2	1.1154	0.3343	0.4135	0.2764	0.1715	**0.0305**	0.0310	U_7	17.9720	3.7608	3.9544	3.8145	1.8463	0.0345	**0.0332**
	U_3	3.4531	0.7923	0.6667	0.6232	0.4158	0.0308	**0.0306**	U_8	23.2303	5.0701	5.2937	5.0037	2.4928	0.0357	**0.0350**
	U_4	5.5952	1.0756	1.1725	1.0225	0.5131	0.0322	**0.0321**	U_9	29.9894	6.3045	6.4840	6.4924	3.1130	0.0374	**0.0355**
	U_5	9.1162	1.8634	1.9079	1.7498	0.9134	**0.0326**	0.0327	U_{10}	36.9910	7.9271	8.0496	7.9273	3.9483	0.0383	**0.0349**
6	U_1	0.4029	0.0847	0.0778	0.0788	0.0412	0.0369	**0.0359**	U_6	14.1641	2.6111	2.5960	2.6110	1.3510	0.2025	**0.1967**
	U_2	1.6556	0.3591	0.3112	0.4249	0.1873	0.0798	**0.0708**	U_7	19.1079	3.4917	3.4817	3.5266	1.8749	0.2284	**0.2169**
	U_3	3.5694	0.6652	0.6872	0.6662	0.3318	**0.0998**	0.1017	U_8	25.3263	4.6177	4.5389	4.5838	2.2612	**0.2573**	0.2647
	U_4	6.2642	1.1898	1.2208	1.1718	0.6135	0.1426	**0.1406**	U_9	31.6424	5.7885	5.7217	5.7526	2.8485	0.2862	**0.2796**
	U_5	9.7829	1.8580	1.7843	1.8281	0.9943	**0.1695**	0.1735	U_{10}	39.4814	7.1958	7.0980	7.1649	3.5601	0.3271	**0.2963**
7	U_1	0.3318	0.1068	0.1117	0.1057	0.0523	**0.0509**	0.0519	U_6	15.5003	4.7125	4.8973	4.6142	2.8466	**0.0551**	0.0563
	U_2	2.3630	0.6663	0.6940	0.4025	0.3108	**0.0528**	0.0533	U_7	21.4162	6.6403	7.1164	6.6993	3.1523	**0.0547**	0.0575
	U_3	3.6491	1.1162	1.4980	1.2518	0.5613	**0.0520**	0.0530	U_8	28.1018	9.0520	9.4185	8.8016	4.6618	**0.0547**	0.0577
	U_4	5.9008	1.9729	1.9632	2.0504	0.9498	**0.0531**	0.0545	U_9	35.8657	11.2129	11.8464	11.3731	5.9470	**0.0563**	0.0592
	U_5	10.1097	3.2351	3.6260	3.2969	1.5419	**0.0532**	0.0543	U_{10}	43.3040	14.0223	14.5950	13.9979	7.1352	**0.0572**	0.0598
8	U_1	1.1740	0.4079	0.4159	0.4009	0.2103	**0.2031**	0.2045	U_6	62.9391	14.0041	13.9084	13.8596	7.1195	1.0801	**1.0774**
	U_2	6.1734	1.5687	1.5919	1.5666	0.7620	**0.3699**	0.3787	U_7	86.6728	19.0267	18.8800	18.7875	10.0050	1.2665	**1.2488**
	U_3	14.8452	3.4008	3.3973	3.4698	1.7864	0.5415	**0.5366**	U_8	114.7233	24.9118	24.7363	24.8067	12.2230	1.4619	**1.4528**
	U_4	27.1244	6.0785	6.0613	6.0709	3.1213	**0.7135**	0.7141	U_9	145.6670	32.8464	32.1774	31.9250	17.2584	**1.6612**	1.6778
	U_5	44.0745	9.5278	9.5214	9.7638	4.7908	0.9077	**0.9057**	U_{10}	184.3387	39.3288	40.3632	40.8053	20.3610	**1.8424**	1.8735
9	U_1	11.4280	3.6820	3.6749	3.6788	1.9920	1.8048	**1.7747**	U_6	522.1018	163.1812	165.0964	162.3380	86.9923	**1.8692**	1.9358
	U_2	80.7595	24.8117	17.3029	21.7718	13.4639	1.7977	**1.7907**	U_7	738.1480	237.7911	239.7231	239.3652	138.3218	**1.9794**	2.0171
	U_3	111.0100	47.0233	35.2241	45.3361	25.1316	**1.8677**	1.8748	U_8	963.4279	311.7575	308.5283	314.5960	175.3361	2.0969	**2.0003**
	U_4	215.2603	66.2718	61.3860	61.5526	35.5612	**1.8531**	1.8826	U_9	1199.0236	392.6148	388.5916	391.1647	201.6610	2.1103	**2.0647**
	U_5	332.3774	114.1196	112.7446	109.4021	61.1653	**1.8374**	1.8675	U_{10}	1489.4470	480.9877	476.7899	477.0074	248.3395	2.2336	**2.0822**

Here, Figure 3 can be obtained by the data in Table 10. From Table 10 and Figure 3, we can get two important results as follows.

(1) SPRS model always takes less time to compute LA of Pawlak RS than TPRS model, which is independent of the scale of universe. And the more data there is, the more prominent the computational efficiency of SPRS is. For example, for the universe U_{10} in the ninth data set, the time consumption of SPRS and TPRS models is 2.0822



(a) (b) (c)

Figure 3: The time consumption for computing LA in different universes

seconds and 480.9877 seconds, respectively. Obviously, there is a gap of approximately 240 times between them.

(2) Regardless of the number of data in universe, compared to all models except LRS, SPRS model always takes less time to obtain the LA. While the computational efficiency of SPRS and LRS is similar, and the size of universe has no significant impact on this result.

Table 11: The time consumption for computing LA in different target concepts

No.s	X	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS	X	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS
1	X_1	0.7319	0.1521	0.1529	0.1539	0.0708	0.0150	**0.0137**	X_6	0.8447	0.1639	0.1676	0.1639	0.0739	0.0950	**0.0640**
	X_2	0.7519	0.1536	0.1559	0.1559	0.0711	0.0280	**0.0270**	X_7	0.8537	0.1629	0.1659	0.1639	0.0742	0.1150	**0.0470**
	X_3	0.7413	0.1561	0.1579	0.1570	0.0713	**0.0430**	**0.0430**	X_8	0.8651	0.1661	0.1699	0.1686	0.0753	0.1331	**0.0310**
	X_4	0.7768	0.1610	0.1629	0.1640	0.0724	0.0615	**0.0611**	X_9	0.9167	0.1659	0.1669	0.1659	0.0795	0.1490	**0.0150**
	X_5	0.8173	0.1631	0.1649	0.1639	**0.0731**	0.0812	0.0790								
2	X_1	4.3714	1.0115	1.0187	1.0197	0.5301	0.0946	**0.0930**	X_6	4.7656	1.0822	1.0876	1.0886	0.5684	0.6373	**0.4154**
	X_2	4.4310	1.0419	1.0468	1.0463	0.5432	0.1995	**0.1989**	X_7	4.7323	1.1045	1.1058	1.1147	0.5763	0.7550	**0.3182**
	X_3	4.6743	1.0495	1.0567	1.0578	0.5486	0.3062	**0.3049**	X_8	4.7018	1.0985	1.1108	1.1078	0.5811	0.8868	**0.2059**
	X_4	4.5873	1.0609	1.0740	1.0730	0.5513	0.4159	**0.4145**	X_9	4.8452	1.1220	1.1353	1.1338	0.5891	1.0118	**0.1040**
	X_5	4.5629	1.0919	1.1044	1.1001	0.5594	0.5270	**0.5258**								
3	X_1	2.8633	0.5536	0.5427	0.5450	0.3003	0.0518	**0.0509**	X_6	3.0481	0.5834	0.5860	0.5926	0.3294	0.3559	**0.2277**
	X_2	2.8830	0.5610	0.5577	0.5610	0.3087	**0.1124**	0.1140	X_7	3.2082	0.6038	0.6092	0.6084	0.3335	0.4169	**0.1616**
	X_3	2.8404	0.5596	0.5637	0.5585	0.3132	**0.1647**	0.1666	X_8	3.3706	0.6014	0.6245	0.6044	0.3396	0.4859	**0.1067**
	X_4	2.9114	0.5757	0.5717	0.5595	0.3185	0.2263	**0.2145**	X_9	3.5347	0.6164	0.6045	0.6260	0.3408	0.5616	**0.0578**
	X_5	2.9335	0.5934	0.5991	0.5974	0.3264	0.2852	**0.2832**								
4	X_1	17.0027	3.9240	3.9235	3.9263	1.7911	0.3741	**0.3683**	X_6	19.4429	4.2517	4.2456	4.2587	2.1273	2.5002	**1.6478**
	X_2	18.0133	4.0533	3.9701	4.0293	1.8098	0.7735	**0.7703**	X_7	20.4778	4.2824	4.2598	4.2671	2.1419	2.9761	**1.2310**
	X_3	18.6551	4.0403	4.1828	4.1444	1.8504	1.1786	**1.1695**	X_8	20.3463	4.3310	4.3538	4.3434	2.1596	3.4081	**0.8225**
	X_4	19.0479	4.0997	4.1354	4.1769	1.9113	1.6073	**1.5976**	X_9	20.4050	4.5681	4.5499	4.5397	2.1675	4.0965	**0.4115**
	X_5	18.6920	4.1962	4.1780	4.2197	2.0815	2.0698	**2.0567**								
5	X_1	30.7627	5.5124	5.5242	5.5295	2.8337	0.5284	**0.5278**	X_6	33.1671	6.1358	6.2748	5.8240	3.1438	3.1667	**2.6481**
	X_2	32.6039	5.6659	5.4821	5.4204	2.8342	1.0544	**1.0508**	X_7	34.5047	6.0835	6.2619	5.9730	3.1498	3.7051	**2.1144**
	X_3	32.3583	5.7222	5.7435	5.6651	2.9357	1.5955	**1.5888**	X_8	36.1980	6.0353	6.4435	6.2167	3.1530	4.2259	**1.5789**
	X_4	32.2079	5.6138	5.9440	5.8385	2.9365	2.1223	**2.1074**	X_9	37.5642	6.2385	6.7772	6.4000	3.1584	4.7518	**1.0658**
	X_5	32.7192	5.8597	6.0084	5.7246	3.0394	**2.6400**	2.6457								
6	X_1	29.0113	8.3836	8.2320	8.2620	4.6714	1.2088	**0.7660**	X_6	34.1967	10.0151	9.9014	8.8605	4.2105	5.4452	**3.9339**
	X_2	35.7733	9.1755	7.8859	6.9175	4.6733	**1.2826**	1.4571	X_7	40.3092	11.1710	9.4409	10.7128	4.0384	6.5520	**2.7059**
	X_3	35.0453	6.7609	6.7979	6.9454	4.6852	2.1842	**1.9967**	X_8	47.9954	13.0692	9.5062	9.5105	4.9431	7.4590	**1.8287**
	X_4	36.0703	9.0179	10.5554	8.3936	4.6877	4.5877	**3.8936**	X_9	49.2934	9.3256	9.3073	9.7312	4.9658	9.8900	**0.8712**
	X_5	35.9687	11.7476	9.6988	11.1228	4.1913	4.9620	**3.7709**								
7	X_1	50.1704	11.1899	11.1159	11.0163	6.2218	1.1084	**1.1041**	X_6	55.7775	11.9894	10.7806	12.3604	6.4953	6.1297	**5.5388**
	X_2	50.1635	11.6925	11.0451	10.9699	6.2384	**2.0615**	2.4735	X_7	56.5947	11.9180	10.5998	12.2829	6.5318	7.1319	**4.5646**
	X_3	52.2432	11.8952	10.9438	11.4947	6.2643	**3.1661**	3.5252	X_8	58.4985	12.2773	11.0206	12.2596	6.5713	8.3816	**3.5133**
	X_4	54.3069	11.7440	10.8478	11.2798	6.2334	4.5208	**4.3841**	X_9	58.4892	12.4897	10.9204	12.3455	6.5106	9.3048	**2.3714**
	X_5	56.1519	11.9410	10.8038	11.7731	6.4316	**5.0747**	5.5963								
8	X_1	171.1221	44.0022	42.8408	43.1613	23.7915	4.2631	**4.1663**	X_6	175.4944	47.4940	47.6369	47.7284	24.6984	27.8994	**17.8496**
	X_2	170.3744	43.3608	43.5247	43.1948	23.6648	8.5301	**8.4258**	X_7	183.4739	47.8719	47.9399	48.1898	24.8912	33.0647	**13.3864**
	X_3	173.0614	45.9732	45.9559	45.7759	23.6194	**13.2615**	13.6250	X_8	181.7856	48.1872	48.4805	48.4289	24.7769	38.5688	**8.8134**
	X_4	179.1732	46.1807	46.3589	46.1769	23.7613	18.1234	**18.0109**	X_9	186.4951	49.2353	48.9710	49.2636	24.1140	44.1506	**4.3554**
	X_5	177.6678	47.0834	46.9392	46.9619	24.4691	23.1530	**22.8880**								
9	X_1	1630.0957	344.4836	341.4052	340.5702	167.3340	35.2002	**34.6222**	X_6	1855.9502	361.2939	377.6293	367.1202	173.6140	193.2551	**173.0095**
	X_2	1673.4235	347.2022	360.6339	350.6613	168.1946	64.5283	**61.2282**	X_7	1928.5680	363.6652	381.9210	379.7134	175.0048	225.8352	**139.1737**
	X_3	1674.3552	356.9558	358.1380	350.2956	169.6648	**96.4172**	103.6314	X_8	2027.6925	367.8221	393.3836	388.7150	175.9762	258.1618	**103.8976**
	X_4	1731.0074	361.2274	357.0676	358.9260	171.3644	**128.4805**	138.6131	X_9	2021.7360	361.6415	412.3720	389.3733	176.2843	290.3578	**69.7588**
	X_5	1763.5885	360.3457	369.8323	370.5823	172.9465	**160.8882**	173.2931								

Note that the LA of SPRS can be obtained by algorithms 3 and 4. According to Table 6, if the number of data in the target concept does not exceed half of the total number of data, we use algorithm s to calculate the LA. Otherwise,

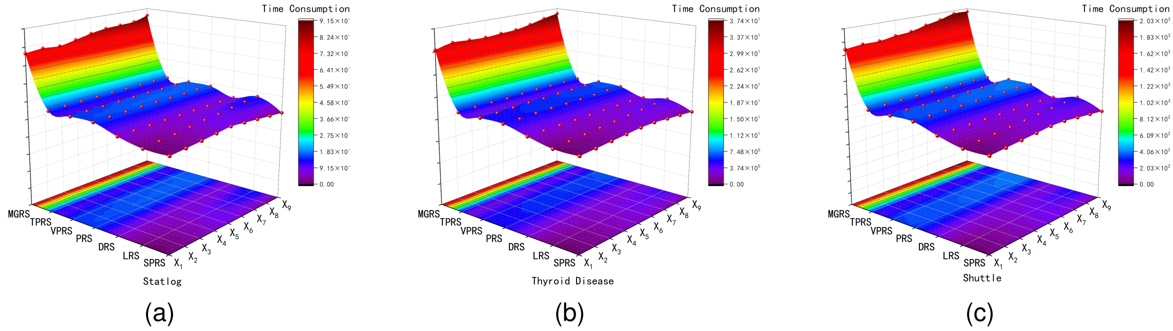


Figure 4: The time consumption for computing LA in different target concepts

algorithm 4 will be selected.

Based on Table 11 and Figure 4, the following results will be obtained.

(1) For any target concept, SPRS is able to calculate LA faster than TPRS. And the efficiency of SPRS can even be more than ten times higher than that of TPRS.

(2) For any target concept, compared to other models except LRS, SPRS always takes less time. If the number of data in the target concept does not exceed half of the number in the data set, SPRS and LRS take similar amount of time to compute the LA. If the scale of the target concept is larger, the calculation speed of SPRS increases rapidly. At this point, the efficiency of SPRS can be about ten times of that of LRS.

• Comparative analysis on attribute reduction

In this section, compared with other traditional RSs, we will verify the efficiency of feature selection with respect to SPRS.

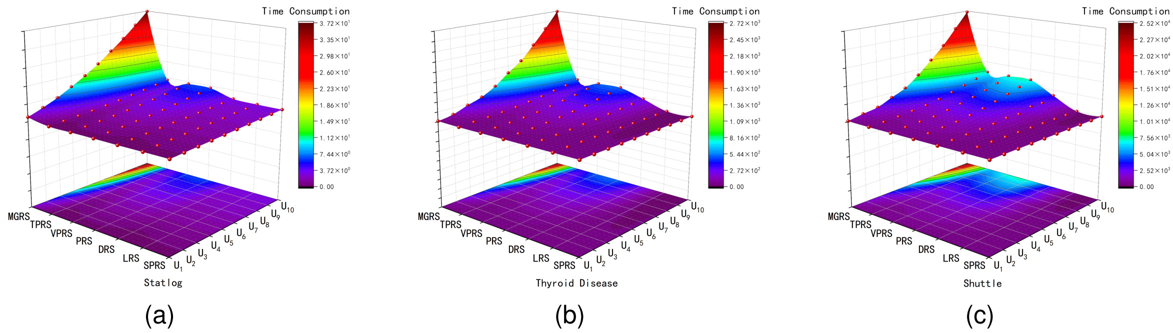


Figure 5: The time consumption for computing reductions $Reduct(AT)_{L,X}$ of RS models in different universes

From Table 12 and Figure 5, we have the following important results.

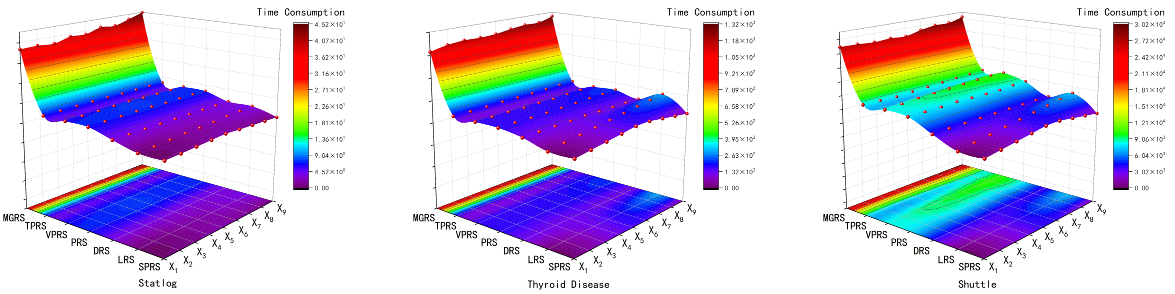
(1) For any universe, calculating $Reduct(AT)_{L,X}$ by using SPRS takes less time than that by using TPRS. Moreover, as the size of the universe increases, the advantage of SPRS becomes more apparent. For example, for the universe U_{10} in the ninth data set, the time taken by TPRS and SPRS is 6754.9284 seconds and 32.9951 seconds, respectively. Obviously, the efficiency of SPRS is more than two hundred times higher than that of TPRS.

(2) No matter how much data the universe contains, we can use the SPRS instead of MGRS, VPRS, PRS and DRS to select the desired attributes or features faster. While for the universe of any size, the efficiency of using SPRS and LRS to select attributes is similar.

Here, we will verify that using SPRS can quickly select attributes in different scales of target concept. According to Table 7, if the scale of the target concept is less than half of the scale of the universe, algorithm 5 will be employed

Table 12: The time consumption for computing reductions $Reduct(AT)_{L,X}$ of RS models in different universes

No.s	U	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS	U	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS
1	U_1	1.9779	0.3359	0.3351	0.3369	**0.2846**	0.3195	0.3185	U_6	17.6356	2.9171	2.9495	2.9419	1.8469	1.8411	**1.8317**
	U_2	4.2842	0.7054	0.7124	0.7104	**0.4515**	0.6173	0.6191	U_7	22.2072	3.7088	3.7474	3.7534	2.2516	**2.1498**	2.1590
	U_3	6.9453	1.1365	1.1407	1.1394	0.9819	0.9165	**0.9145**	U_8	26.8004	4.5032	4.5589	4.6306	2.5849	2.5455	**2.4472**
	U_4	10.2135	1.6430	1.6629	1.6679	1.2222	1.2312	**1.2168**	U_9	31.9407	5.4503	5.4640	5.4384	2.9915	2.7805	**2.7724**
	U_5	13.6863	2.2478	2.2805	2.2429	1.6051	**1.5310**	1.5431	U_{10}	37.3944	6.3865	6.4437	6.4351	3.5163	3.1253	**3.0499**
2	U_1	100.6286	20.2101	20.1839	20.0377	**13.2169**	19.4126	19.2153	U_6	977.3303	201.3501	201.5366	203.0291	135.1630	**116.4785**	118.0968
	U_2	227.9829	45.4617	45.7109	45.6978	39.0519	**38.5437**	38.8345	U_7	1232.8326	252.6657	252.7511	253.0056	158.3361	138.7830	**137.4041**
	U_3	382.8443	76.5307	77.1198	77.4911	59.3318	**58.5431**	58.5879	U_8	1517.3755	309.4786	310.5106	310.8196	194.3310	**158.2086**	158.6285
	U_4	555.9708	113.0485	114.2998	114.1692	84.9493	78.4673	**78.2905**	U_9	1783.3055	371.4025	368.7235	371.2445	212.7920	177.0298	**176.8607**
	U_5	759.2833	153.9150	155.1204	157.1008	107.9964	**97.7969**	98.4999	U_{10}	2109.7167	433.9403	433.1682	436.4988	237.9943	**195.8267**	196.9899
3	U_1	2.3448	0.3845	0.3692	0.3835	**0.2120**	0.3182	0.3251	U_6	32.4027	5.3299	5.3697	5.3578	2.6618	1.8810	**1.8701**
	U_2	5.9820	0.9581	0.9375	0.9305	0.6505	0.6933	**0.6263**	U_7	42.5744	6.9237	6.9112	6.9863	3.7418	**2.1773**	2.2650
	U_3	10.8655	1.7318	1.7447	1.7043	0.9518	0.9473	**0.9327**	U_8	53.2691	8.6211	8.6415	8.6586	4.3695	2.5098	**2.4596**
	U_4	16.8820	2.7202	2.7017	2.8146	1.4582	**1.2558**	1.2880	U_9	65.5783	10.6539	10.6074	10.6015	5.9521	**2.7826**	2.8167
	U_5	24.3041	3.9316	3.8943	3.9962	1.6485	**1.5570**	1.5850	U_{10}	78.6739	12.9308	12.8763	12.8952	7.0031	3.1522	**3.0708**
4	U_1	7.1124	1.3868	1.4027	1.4008	1.1416	1.0909	**1.0895**	U_6	117.8792	23.6177	23.6145	23.6868	12.3628	6.4589	**6.4405**
	U_2	19.7804	3.8887	3.7980	3.8720	2.2015	2.1983	**2.1702**	U_7	152.6920	31.2871	31.1298	31.3798	17.3361	**7.4211**	7.5341
	U_3	36.8090	7.2988	7.2948	7.4388	3.3594	**3.2690**	3.2764	U_8	195.9262	39.6968	39.3769	39.5869	22.3138	**8.6261**	8.6863
	U_4	59.3925	11.5999	11.6955	11.7943	5.9352	4.3531	**4.3153**	U_9	240.0504	49.4151	49.1823	49.8309	27.9634	**9.6458**	9.6726
	U_5	86.0120	16.9463	16.7618	17.4464	8.9920	**5.3981**	5.3997	U_{10}	291.0619	59.8983	59.0401	59.4777	31.6923	10.7743	**10.6920**
5	U_1	20.6443	3.5360	3.6239	3.6083	**1.9253**	2.6577	2.6569	U_6	975.8524	164.4130	171.9525	163.8341	83.3195	**2.7432**	2.8917
	U_2	96.6143	22.0668	18.9024	21.0557	11.3816	2.7743	**2.7294**	U_7	1275.1843	224.4869	235.0553	225.9584	123.3284	**2.9064**	2.9454
	U_3	264.5088	41.0548	36.9027	39.7906	21.0390	2.7219	**2.6764**	U_8	1714.9532	301.0812	305.8039	307.3761	156.3294	**2.8555**	2.9390
	U_4	405.9342	70.7401	64.3878	72.7310	36.0417	2.7972	**2.7752**	U_9	2181.5346	371.2047	392.9681	385.5737	189.9581	2.9795	**2.9148**
	U_5	643.6298	111.4719	106.3475	112.7508	66.1305	**2.7496**	2.9021	U_{10}	2689.2521	468.3123	469.4291	474.9613	253.4916	2.9716	**2.9401**
6	U_1	12.8167	2.4565	2.4684	2.4943	1.8284	2.1592	**1.7087**	U_6	270.9167	49.8823	50.8008	50.6991	26.3185	10.7799	**10.4591**
	U_2	40.5705	8.3469	7.8041	7.7606	4.1862	**3.6922**	3.7688	U_7	365.1541	68.0164	68.4200	74.3691	35.0054	**12.3654**	13.0020
	U_3	80.9539	15.1405	15.4318	15.3322	8.3943	5.6501	**5.5185**	U_8	473.6454	86.8257	88.4165	87.5879	45.2196	**14.1692**	14.2394
	U_4	131.9679	25.4511	26.4354	29.7980	15.7267	**7.3514**	8.5615	U_9	584.4860	121.9329	124.1139	120.4576	68.7492	**16.7660**	17.6591
	U_5	197.0504	36.1992	36.9141	38.5243	19.5299	**8.7986**	8.8738	U_{10}	715.0972	144.9191	135.7756	140.0772	79.9354	**17.9041**	18.6892
7	U_1	16.7028	3.6453	3.6167	3.6122	2.6035	2.5894	**2.5577**	U_6	761.5143	164.0647	166.2291	167.3956	87.9463	3.1818	**3.0035**
	U_2	124.6810	15.3193	18.1436	21.7701	8.9150	2.7373	**2.7078**	U_7	1089.9647	232.8319	223.4652	227.3333	143.1963	3.2594	**2.9563**
	U_3	195.1641	48.0178	43.6077	39.8954	23.3715	2.8849	**2.8254**	U_8	1414.8439	312.1140	305.5933	300.6888	168.3619	3.2701	**3.1131**
	U_4	283.7433	71.1940	67.3844	65.0703	37.3945	2.9010	**2.8018**	U_9	1766.4422	382.6400	388.1786	384.2412	199.1846	3.2200	**3.1008**
	U_5	527.2490	115.7854	108.3916	112.2698	71.6943	3.0640	**2.8904**	U_{10}	2187.3225	475.9421	475.5046	474.7412	251.1846	3.3784	**3.1325**
8	U_1	30.9783	6.7044	7.0081	6.9521	4.7831	5.0569	**4.5411**	U_6	393.9685	97.3524	99.3949	97.3398	48.3608	5.9950	**5.3016**
	U_2	89.6357	17.7656	18.4361	20.3665	8.9258	5.1388	**4.7319**	U_7	488.6349	142.3664	145.3918	142.3145	72.5492	6.0206	**5.5639**
	U_3	143.8557	30.4397	31.4919	34.6971	16.6662	5.3954	**4.8391**	U_8	580.3518	193.6324	199.3916	194.3125	99.0876	6.1235	**5.8638**
	U_4	214.9678	46.1064	48.1973	48.3696	24.1825	5.6421	**5.0135**	U_9	691.3495	268.1507	275.8835	271.6351	137.9635	6.3597	**6.1101**
	U_5	283.5630	68.3429	69.7648	72.4397	35.0051	5.8499	**5.1394**	U_{10}	814.6642	377.4193	384.3662	379.3334	194.8053	6.6327	**6.2934**
9	U_1	187.7466	52.1803	51.9674	52.3225	28.9915	28.7635	**28.4988**	U_6	8406.0183	2330.0022	2355.2028	2419.2097	1184.0925	33.1301	**31.2958**
	U_2	1072.1137	192.6647	337.7399	259.6502	85.9302	29.9961	**28.6983**	U_7	12033.6405	3354.0526	3259.1054	3347.7882	1125.2919	33.5955	**32.2933**
	U_3	1850.0787	555.2914	601.0041	614.8871	276.0310	31.7183	**29.4750**	U_8	15713.1725	4456.5791	4288.2954	4484.7732	2194.3308	33.4276	**32.2024**
	U_4	3635.8027	1045.2326	882.6127	939.8061	486.1258	32.7094	**29.3204**	U_9	20195.0603	5571.7179	5444.6125	5580.2068	2126.2915	35.3680	**31.9831**
	U_5	5413.6783	1590.1432	1611.5565	1624.5019	749.3633	**32.3746**	30.6001	U_{10}	24729.2719	6754.9284	6743.0026	6772.4598	3334.1389	37.0082	**32.9951**



(a) (b) (c)

Figure 6: The time consumption for computing reductions $Reduct(AT)_{L,X}$ of RS models in different target concepts

to select attributes more quickly. Otherwise, algorithm 6 will be used. From Table 13 and Figure 6, we have the following basic facts.

(1) For target concepts containing any amount of data, the efficiency of using SPRS to calculate reduction is higher than that of TPRS. Specifically, when the scale of the target concept is particularly small or large, the advantage of SPRS will be greater.

(2) For target concepts containing any amount of data, compared to models such as MGRS, VPRS, PRS, and DRS, SPRS can be used to select the required attributes more quickly. If the size of the target concept does not exceed half of that of the data set, both SPRS and LRS can be used to quickly select attributes, and the speed of attribute selection

Table 13: The time consumption for computing reductions $Reduct(AT)_{L,X}$ of RS models in different target concepts

No.s	X	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS	X	MGRS	TPRS	VPRS	PRS	DRS	LRS	SPRS
1	X_1	36.6976	6.3570	6.4181	6.4633	3.2715	0.3187	**0.2415**	X_6	40.5543	6.6884	6.7518	6.6676	3.3548	2.1379	**1.3362**
	X_2	36.9369	6.3971	6.5099	6.5072	3.2944	**0.5356**	0.5691	X_7	40.0751	6.7108	6.7314	6.7749	3.3394	2.6808	**0.9988**
	X_3	36.4261	6.5999	6.6413	6.6659	3.3087	0.9684	**0.9661**	X_8	42.3422	6.7617	6.7369	6.7927	3.3692	2.9979	**0.6440**
	X_4	38.5049	6.6767	6.9831	6.8222	3.3490	1.5381	**1.4029**	X_9	44.8270	6.7547	6.7460	6.7251	3.3947	3.4798	**0.2715**
	X_5	38.2665	6.8089	7.0607	6.9301	3.3615	**1.7199**	1.8328								
2	X_1	1558.3150	353.0747	318.7862	318.5877	170.3321	33.6243	**32.4814**	X_6	1754.5069	389.9895	351.3492	343.8250	180.9940	201.6716	**162.9441**
	X_2	1566.5049	354.4419	322.2124	324.9994	170.9161	66.9818	**64.8106**	X_7	1743.7193	411.4860	369.7579	361.4088	181.9463	235.6558	**129.6087**
	X_3	1650.3223	354.5530	339.9732	340.9205	172.2494	100.8934	**97.9332**	X_8	1765.9133	435.2008	375.8752	358.0608	183.9466	268.7840	**97.7098**
	X_4	1657.3133	364.5605	347.9727	335.7244	177.1699	134.5093	**130.4178**	X_9	1757.5781	452.3873	382.8376	372.6655	188.1993	302.7991	**64.7813**
	X_5	1737.2826	378.7818	345.7079	333.5758	179.1386	168.3706	**162.9361**								
3	X_1	76.9023	12.8012	13.0222	13.1424	7.8815	**3.6059**	3.7072	X_6	84.3162	13.8513	14.0753	14.0545	7.0845	9.1126	**6.7749**
	X_2	75.9403	13.2355	13.2290	13.1110	7.1182	4.7590	**4.6737**	X_7	89.3120	13.9320	13.9592	13.9304	7.2193	10.4728	**5.7985**
	X_3	78.8868	13.2508	13.1984	13.3640	7.0039	5.7359	**5.7318**	X_8	90.8558	14.0185	14.2639	14.3055	7.1547	11.7073	**4.6466**
	X_4	80.2481	13.5575	13.3149	13.3389	7.9715	6.8240	**6.7746**	X_9	94.6658	14.0827	14.4346	14.3467	7.2039	13.1698	**3.6680**
	X_5	80.8297	13.8663	13.8763	13.7970	7.9736	8.2251	**7.9095**								
4	X_1	285.2071	58.4628	60.5482	59.5794	31.8463	**13.0875**	13.2844	X_6	342.8252	63.0736	62.3146	63.2298	34.1659	40.3239	**29.0808**
	X_2	284.2664	60.2019	59.7857	60.5526	32.0847	18.2386	**18.1849**	X_7	336.6062	63.5545	63.4349	64.0174	34.2510	47.1661	**24.0633**
	X_3	300.2404	62.6885	61.5584	62.1608	32.8431	**23.7387**	23.7922	X_8	343.3722	64.1137	63.8781	64.7815	34.3371	52.7976	**18.9477**
	X_4	316.5769	61.9780	62.5330	62.8432	33.6492	29.4428	**29.0348**	X_9	353.9865	67.6782	67.3194	67.2840	35.5208	61.7055	**13.8391**
	X_5	329.6314	62.3464	62.3722	63.0661	34.1158	34.8761	**34.0965**								
5	X_1	1103.1636	190.0786	188.3967	188.7906	137.1884	34.1252	**33.7407**	X_6	1233.1709	212.4663	194.1113	203.5311	146.4746	204.5030	**145.6793**
	X_2	1160.0605	186.9908	188.1185	185.9338	139.5846	68.2179	**67.5618**	X_7	1260.2580	218.6874	205.0760	212.9481	148.1515	238.7675	**134.7203**
	X_3	1184.5200	193.6015	190.7185	183.9317	141.2895	**102.3790**	101.4878	X_8	1260.9376	231.5135	213.4035	216.6183	148.9969	272.9597	**100.8924**
	X_4	1181.6863	203.3043	187.1390	189.5278	143.0039	136.3649	**135.0409**	X_9	1330.2879	231.1615	211.3668	229.4246	149.1188	306.9996	**67.3597**
	X_5	1194.5794	206.0728	195.0831	196.2439	**154.3945**	170.3137	169.2146								
6	X_1	855.4559	160.0473	165.9921	178.2074	77.1149	**18.6897**	19.2900	X_6	827.4491	182.7963	183.2563	182.6594	80.8462	52.7042	**37.2372**
	X_2	861.9443	162.7748	178.9630	179.1662	78.3210	**20.8384**	20.8798	X_7	829.2313	186.4313	186.8101	187.4526	80.4467	62.9507	**29.5238**
	X_3	848.7194	173.4052	180.2783	180.5365	78.9748	**28.3523**	28.6860	X_8	830.1124	186.2462	187.5965	187.6538	79.4823	70.7800	**22.2942**
	X_4	840.5062	174.3291	180.1227	180.9620	79.1185	36.3034	**36.0996**	X_9	835.3485	188.6488	187.2941	188.2406	81.1118	80.4747	**18.8348**
	X_5	844.4260	179.3587	181.7937	181.5853	79.4973	44.1849	**44.1034**								
7	X_1	1064.8504	214.6636	213.4953	222.7989	149.8846	34.7949	**34.3372**	X_6	1169.4624	227.0727	231.8097	246.8540	166.1482	211.3620	**161.4476**
	X_2	1076.7798	212.4903	221.9402	226.0558	152.3394	**69.2732**	74.8744	X_7	1226.3268	240.2356	238.9211	255.8232	157.1184	244.7919	**139.1706**
	X_3	1113.6156	216.4481	221.3506	229.1921	153.1984	**103.4834**	109.2443	X_8	1205.3031	249.9577	239.1399	259.3642	159.1365	286.7189	**110.1145**
	X_4	1133.6313	223.9889	225.3332	241.7506	157.1195	141.3913	**139.7951**	X_9	1197.1494	264.2096	250.0322	271.6758	162.8432	316.5978	**68.3396**
	X_5	1192.9256	224.7330	221.8434	237.9962	**164.2949**	175.4007	179.1545								
8	X_1	1958.4032	435.6568	437.9733	439.5902	287.1184	60.2578	**59.7765**	X_6	2178.4571	503.7325	495.6582	532.5266	300.1846	360.3527	**299.4760**
	X_2	2035.8204	458.8927	450.3599	463.2823	290.1846	**123.6486**	125.7950	X_7	2178.0859	525.5329	510.7796	556.7030	303.9160	434.3821	**238.7361**
	X_3	2152.7371	453.0280	445.7574	479.7848	293.2250	188.4903	**185.0951**	X_8	2268.2912	541.8812	537.0639	551.6403	304.8483	480.5918	**191.3652**
	X_4	2204.1050	469.2585	463.2361	492.5314	295.9358	250.7031	**244.5664**	X_9	2283.0185	552.6304	534.2221	577.1697	309.8318	552.5698	**120.6071**
	X_5	2222.7111	481.0586	473.8532	509.6513	297.9997	309.4499	**297.3803**								
9	X_1	27316.4933	8492.3923	8645.3194	8162.5904	4301.9465	832.4462	**813.3551**	X_6	28223.2803	9163.7391	10247.2112	8149.1764	4449.4948	4988.1445	**4065.1261**
	X_2	27180.1935	8337.7764	8957.1113	8051.9014	4372.6314	1675.9883	**1639.5053**	X_7	29078.0570	9543.6098	10482.8324	8077.2799	4455.1795	5819.0437	**3261.9712**
	X_3	26817.8140	8185.6473	9438.7109	8129.3370	4367.4968	2491.5197	**2438.1044**	X_8	29104.8873	9786.1992	10697.5849	8537.8249	4489.6330	6675.5628	**2449.4604**
	X_4	27163.1290	8610.9773	9694.5904	8409.8733	4395.6540	3329.3511	**3258.1633**	X_9	30578.5750	9992.9319	10517.2861	8597.4188	4396.4487	7496.7497	**1633.8596**
	X_5	26876.4928	8889.4180	9949.4720	8263.6673	4369.1182	4164.5351	**4061.5215**								

is similar. However, if there is a large amount of data in the target concept, the speed of selecting attributes by using SPRS is significantly faster than that by using LRS.

Here, all comparative results between SPRS and other models are summarized in Table 14. The letter H indicates that the computational efficiency of SPRS model is higher than that of the compared model. The letter S means that the computational efficiency of SPRS model and the compared model is similar. And the sizes of universe and target concept are denoted by $|U|$ and $|X|$, respectively. From Table 14, regardless of the sizes of U and X, the efficiency of SPRS in calculating UA is always higher than that of other models. When calculating LA and attribute reduction, two conclusions can be drawn: First, for any U and X, the efficiency of SPRS is always significantly higher than that of TPRS, MGRS, VPRS, PRS and DRS. Second, regardless of whether the size of U is larger or smaller or the size of X is smaller, the speed of SPRS and LRS analyzing data is similar; When X is large in scale, SPRS is much faster than LRS in mining data. In summary, SPRS is more effective in analyzing data than all other models.

7. Conclusion

So far, although many RS models have been proposed, most of them are developed based on Pawlak rules. Therefore, the mathematical structures of most RS models are similar, and the computational efficiency of RS methods has not been significantly improved. The main conclusions of this paper include:

(1) TPRS can be simplified and equivalently redesigned based on some new rules. Experimental analysis demonstrates that the computational efficiency of SPRS is extremely excellent, usually several hundred times higher than that of TPRS. Compared with several representative existing RSs, one can find that in most cases, SPRS can analyze data faster than most RSs.

(2) The attribute reductions of UA and LA play an important role in RS theory. In this article, it is proven that the LA reduction for any target concept is equal to the UA reduction of the complement set of that target concept. It

Table 14: Comparative analysis between SPRS and other RS models

Measures	RS models	$\|U\|$ is smaller	$\|U\|$ is larger	$\|X\|$ is smaller	$\|X\|$ is larger
Upper approximation	TPRS	H	H	H	H
	MGRS	H	H	H	H
	VPRS	H	H	H	H
	PRS	H	H	H	H
	DRS	H	H	H	H
	LRS	H	H	H	H
Lower approximation	TPRS	H	H	H	H
	MGRS	H	H	H	H
	VPRS	H	H	H	H
	PRS	H	H	H	H
	DRS	H	H	H	H
	LRS	S	S	S	H
Attribute reduction	TPRS	H	H	H	H
	MGRS	H	H	H	H
	VPRS	H	H	H	H
	PRS	H	H	H	H
	DRS	H	H	H	H
	LRS	S	S	S	H

means that these two reductions are essentially the same.

Fortunately, according to the rules proposed in this paper, most existing rough set models can be equivalently redesigned. This will greatly improve the computational efficiency of RS theory in processing various data problems. This discovery will change people's inherent views on the computational efficiency of RS for over forty years, and effectively promote the development of RS theory.

CRediT authorship contribution statement

Qingzhao Kong: Conceptualization, Investigation, Methodology, Validation, Writing-original draft, Writing-review & editing. **Conghao Yan**: Data curation, Software, Validation. **Weihua Xu**: Funding acquisition, Project administration, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (No. 62376229) and Natural Science Foundation of Chongqing (No. CSTB2023NSCQ-LZX0027).

References

References

[1] E. Eirola, A. Lendasse, V. Vandewalle, et al., Mixture of gaussians for distance estimation with missing data, Neurocomputing 131 (2014).

[2] Y. Qi, Y. Miche, E. Eirola, et al., Regularized extreme learning machine for regression with missing data, Neurocomputing 102 (2013) 45-51.

[3] L. Zadeh, Fuzzy sets, Inf. Control. 8 (1965) 338-353.

[4] W.H. Xu, D.D. Guo, Y.H. Qian, et al., Two-way Concept-cognitive Learning Method: A Fuzzy-based Progressive Learning, IEEE Trans. Fuzzy Syst. 31 (2023) 1885-1899.

[5] X.Y. Zhang, J.R. Li, Incremental feature selection approach to interval-valued fuzzydecision information systems based on λ-fuzzy similarity selfinformation, Inf. Sci. 625 (2023) 563-619.

[6] C.L. Zhang, E.C.C. Tsang, W.H. Xu, et al., Dynamic updating variable precision three-way concept method based on two-way concept-cognitive learning in fuzzy formal contexts, Inf. Sci. 655 (2024) 119818.

[7] B. Zhang, L. Zhang, Theory and Applications of Problem Solving, NorthHolland Publishing, Amsterdam, 1992.

[8] Z. Pawlak, Rough set, Int. J. Comput. Inf. Sci. 11 (1982) 341-356.

[9] Q.Z. Kong, X.W. Zhang, W.H. Xu, et al.,, Attribute reducts of multi-granulation information system, Artif. Intell. Rev. 53 (2020) 1353-1371.

[10] Q.Z. Kong, W.H. Xu, The comparative study of covering rough sets and multi-granulation rough sets, Soft Comput. 23 (2019) 3237-3251.

[11] Y.H. Qian, H.H. Cheng, J.T. Wang, et al., Grouping granular structures in human granulation intelligence, Inf. Sci. 382383 (2016) 150-169.

[12] Y.H. Qian, S.Y. Li, J.Y. Liang, et al., Pessimistic rough set based decisions: A multigranulation fusion strategy, Inf. Sci. 264 (6) (2013) 196-210.

[13] Y.H. Qian, J.Y. Liang, Y.Y. Yao, et al., MGRS: A multi-granulation rough set, Inf. Sci. 180 (6) (2009) 949-970.

[14] Y.H. Qian, J.Y. Liang, C.Y. Dang, Incomplete multigranulation rough set, IEEE Trans. Sys. Man Cybern.: Part A 40 (2) (2009) 420-431.

[15] R.W. Swiniarski, A. Skowron, Rough set methods in feature selection and recognition, Pattern Recogn. Lett. 24 (6) (2003) 833-849.

[16] W.H. Xu, M. Huang, Z.Y. Jiang, et al., Graph-Based Unsupervised Feature Selection for Interval-Valued Information System, IEEE Trans. Neural Netw. Learn. Syst. 2023, https://doi.org/10.1109/TNNLS.2023.3263684.

[17] W.H. Xu, Y.F. Yang, Matrix-based feature selection approach using conditional entropy for ordered data set with time-evolving features, Knowl.-Based Syst. 279 (2023) 110947.

[18] W. Zhu, Topological approaches to covering rough sets, Inf. Sci. 177 (6) (2007) 1499-1508.

[19] W. Zhu, F.Y. Wang, On three types of covering rough sets, IEEE Trans. Knowl. Data Eng. 19 (8) (2007) 1131-1144.

[20] D.D. Guo, W.H. Xu, Fuzzy-based concept-cognitive learning: An investigation of novel approach to tumor diagnosis analysis, Inf. Sci. 639 (2023) 118998.

[21] W.H. Xu, Y.T. Guo, Generalized multigranulation double-quantitative decision-theoretic rough set, Knowl.-Based Syst. 105 (2016) 190-205.

[22] Z. Pawlak, Information systems theoretical foundations, Inform. Syst. 6 (1981) 205-218.

[23] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishers, Boston, 1991.

[24] X.D. Yue, Y.F Chen, D.Q. Miao, et al., Tri-partition neighborhood covering reduction for robust classification, Int. J. Approx. Reason. 83 (2017) 371-384.

[25] R. Jensen, S. Qiang, Semantics-preserving dimensionality reduction: rough and fuzzy-rough-based approaches, IEEE Trans. Knowl. Data Eng. 16 (12) (2004) 1457-1471.

[26] Y.Y. Yao, Interpreting concept learning in cognitive informatics and granular computing, IEEE Trans. Syst. Man. Cybern. B 39 (4) (2009) 855-866.

[27] W. Ziarko, Variable precision rough set model, J. Comput. Syst. Sci. 46 (1) (1993) 39-59.

[28] Z. Pawlak, S.K.M. Wong, W. Ziarko, Rough sets: probabilistic versus deterministic approach, Int. J. Man-Mach. Stud. 29 (1988) 81-95.

[29] Y.H. Qian, X.Y. Liang, Q. Wang, et al., Local rough set: a solution to rough data analysis in big data, Int. J. Approx. Reason. 97 (2018) 38-63.

[30] Q.Z. Kong, W.T. Wang, W.H. Xu, et al., A method of data analysis based on division-mining-fusion strategy, Inf. Sci., 666 (2024) 120450.