ISSN 2616-5775 Vol. 5, Issue 14: 11-21, DOI: 10.25236/AJCIS.2022.051402

Parallel incremental update of semi-monolayer covering rough set based on object set changes

Zhizhong Long[1,a,*], Chenggong Lv[1,b]

[1]*College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo City, China*
[a]*1581095018@qq.com*, [b]*212009020038@home.hpu.edu.cn*
[*]*Corresponding author*

Abstract: *The semi-monolayer covering rough set has the characteristics of high approximation quality and efficient computation. In the face of massive data changes, the study of rough sets will face the challenge of high complexity computation. To address the large-scale changes of object sets in set-valued decision information systems, this paper proposes a dynamic update method based on Spark framework to solve the approximation set problem of semi-monolayer covering rough set. This method is mainly embodied in four aspects: firstly, for the change of information grains in the proposed single-layer covering rough set, this paper proposes an update strategy for the change of information units; then for the update of reliable elements, an update strategy is proposed for* $\underline{C}_{GC0}'(X)$ *and* $\overline{C}_{GC0}'(x)$*; secondly, for the change of reliable elements and disputed elements, an RSM matrix update strategy; and finally the update strategy of the approximation set is proposed for the changes of information units. Since the incremental methods using other models will yield different approximation sets and the prerequisite of the comparison experiments requires the approximation sets to be consistent, in order to ensure the correctness of the experimental results, this paper designs the comparison experiments of static and dynamic algorithms based on the semi-monolayer covering rough set model. The experimental results with several data sets show that the incremental algorithm speeds up in 3.67~7.72 times than the static algorithm when there is an object set update.*

Keywords: *semi-monolayer covering rough set; parallel computation; approximation rules; set value information System; grain calculation*

1. Introduction

Rough set theory (RST) is a mathematical model proposed by Pawlak[1] for effectively dealing with uncertainty, inconsistency, and incompatibility. Currently, rough set theory is widely used in data mining[2], artificial intelligence, image processing, and other fields. It is also known as one of the three main models of granularity computing[3]. Although existing algorithms for rough set knowledge discovery have been very successful in dealing with small as well as medium-sized datasets, they have encountered bottlenecks in dealing with massive datasets and dynamic datasets of large size. Repeated methods for computing approximate sets mostly adopt the batch computing model, i.e., computing the entire current dataset at one time. Facing the huge scale of massive data, the existing methods often take a lot of running time and cannot even be executed under the current software and hardware environment. Rough set-based knowledge discovery methods face the difficulty of high-complexity computation when facing large-scale complex real-world problems. Therefore, how to improve the efficiency of rough set-based knowledge discovery algorithms to fully reflect the advantages of rough sets in solving uncertainty problems has become one of the main tasks in the field of rough set research. On the one hand, the HDFS distribution in Hadoop can store large-scale data, while the parallel computing of Spark framework can greatly reduce the computation time[4]. On the other hand, the incremental knowledge update method can effectively improve the efficiency of knowledge update due to its ability to make full use of the existing knowledge. Therefore, many scholars have devoted themselves to improving the efficiency of knowledge discovery based on rough sets by using parallel computing and incremental knowledge update methods.

Currently, research based on incremental learning of rough sets is found to be used in either complete or incomplete information systems to solve the problems of rule extraction update, attribute simplification and approximation set update.Luo et al[5] studied the update nature of the dynamically maintained approximation values in set-valued decision systems as the criterion values evolve over time, giving the criteria for addition and deletion, respectively, and the experiments were validated in terms of changes in the set of objects, changes in the set of attributes and the effect of object set changes on attribute set changes are verified experimentally.Xu et al[6] proposed a stream computational learning method based on the existing incremental learning of the probabilistic rough set model.The direction addresses both the approximate set changes brought about by adding and deleting object sets, and also shows experimentally that the stream computational learning method is an effective strategy for computing big data.Xie et al[7] studied the incremental The framework of attribute approximation algorithm, Xie et al introduced the concept of inconsistency in incomplete decision systems in the face of attribute value changes and proved that inconsistency-based attribute approximation is equivalent to positive region-based attribute approximation.Hu et al[8] investigated the dynamic mechanism of updating multi-grained rough set approximations during refinement or coarsening of attribute values based on the multi-grained rough sets. Based on rough set theory and granularity computation, Bu et al[9] focused on incremental knowledge discovery methods for dealing with big data.Dong et al[10] studied two incremental algorithms based on when attributes are added versus when they are removed under probabilistic rough sets for solving updates under attribute set changes.Wu et al[11] studied advances related to attribute parsimony in the context of big data.Zhang et al[12] focused on a matrix approach based on rough sets to accomplish incremental update of approximate sets and applied it to a parallel rough set approximation computation method for large-scale data analysis.

Wu et al[13] proposed the semi-monolayer covering rough set (SMC) on the basis of the general covering rough set model. Then four pairs of upper and lower approximation operators in the form of granularity are defined in the SMC approximation space, revealing the equivalence between the point-shaped approximation operator and the granular approximation operator, resulting in a granular algorithm for computing the approximation set and improving the computational efficiency of the approximation set.Wu et al[14] improve the approximation operator of the semi-monolayer covering by replacing the whole block with a reliable set, which is easier to granularize and compute efficiently than before. In addition, the semi-monolayer covering rough set is between general covering and division and represents a special representative covering, but it cannot handle dynamic set-valued information systems. Therefore, when the information system changes in real time, the approximation set needs to be recomputed.

The semi-monolayer covering rough set based on granular computation has the characteristics of high approximation quality approximation set and efficient computation, but the problem of dynamic updating of approximation set in Spark framework has not been studied. Therefore, in this paper, we study the problem of dynamic update of the approximation set of the semi-monolayer covering rough set when the argument domain changes in the set-valued information system under the Spark framework. The semi-monolayer covering is able to decompose the information grain into reliable and disputed units according to whether the information value contains a set value or not. When the set of objects in the set-valued system is migrated, the number of reliable and disputed units in the set-valued information system will change, which will lead to a change in the approximation set. When an object is added to the original system, the possible changes of information units in the proposed single-layer overlay are discussed by case, and then by analyzing the effect of information unit changes on $\mathrm{RS}(\mathrm{Cell}_c)$ and designing the related update strategy, finally the change of $\mathrm{RS}(\mathrm{Cell}_c)$ will lead to the update of $DA0'$ and $DE0'$ approximation sets in the system change. In addition, a static algorithm and a corresponding incremental algorithm were developed. Finally, the effectiveness of the algorithm was verified by conducting experiments on the UCI dataset.

2. The semi-monolayer covering rough set

The set-valued decision information system (SVDIS) consists of a quadruplet of $(\mathrm{U}, \mathrm{A} \cup \mathrm{D}, \mathrm{V}, \mathrm{f})$, $\mathrm{A} = \{\mathrm{a}_1, \mathrm{a}_2, \mathrm{a}_3, \cdots, \mathrm{a}_\mathrm{n}\}$, A is a non-empty conditional attribute, D is a decision attribute, where $\mathrm{A} \cap \mathrm{D} = \emptyset$,V represents the value domain of the attribute, then $\mathrm{V} = \bigcup_{\mathrm{a_i} \in \mathrm{A}} \mathrm{V}_{\mathrm{a_i}} \cup \mathrm{V}_\mathrm{d}$, $\mathrm{f}: \mathrm{U} \times \{\mathrm{A} \cup \mathrm{D}\} \rightarrow 2^\mathrm{V}$ represents the set-valued mapping from U to V[15].

Definition 1.[14]. Let $\mathrm{S} = (\mathrm{U}, \mathrm{A} \cup \mathrm{D}, \mathrm{V}, \mathrm{f})$ be a set-valued decision information system.

If each attribute value in $\overrightarrow{\mathrm{Cell}}$ is a single value, then this Cell is reliable and is denoted by Cell_r. The set of all reliable information units in CELL is denoted by RC and $\overrightarrow{\mathrm{RC}}$is the set of reliable meta-information interpretations.

If $\overrightarrow{\mathrm{Cell}}$ there exists an attribute whose value is a set value, then the current Cell is unreliable, denoted by Cell_c, and the set of all unreliable information cells in CELL is denoted by CC, $\overrightarrow{\mathrm{CC}}$ the set of information interpretation of the disputed cells.

Definition 2[14] Let $S = (U, A \cup D, V, f)$ be a set-valued decision information system. The set of associated reliable elements of $Cell_c$ is denoted as $RS(Cell_c) = \{\overrightarrow{Cell_c} | a_i \in A, x \in Cell_x, y \in Cell_y, f(x, a_i) \subseteq f(y, a_i)\}$, and $Cell_c$ is in one-to-one correspondence with $RS(Cell_c)$ where $RS(Cell_c)$ denotes the relationship between $Cell_c$ and $Cell_r$; the relationship matrix RSM between $Cell_r \in RC, Cell_c \in CC$ can be regarded as the set of key-value pairs with $\overrightarrow{Cell_c}$ as the key and $RS^t(Cell_c)$ as the value $\{Cell_c, RS^t(Cell_c) | \quad Cell_c \in CC\}$.

Definition 3[14] Assuming that (U, C) is a proposed single-layer covering approximation space over a set-valued information system, $Cell_r \in RC$ and $Cell_c \in CC$, and any $X \subseteq U$, the representation of the information cell-based approximation set is shown below:

$$\underline{C}_{DAO}(X) = \cup (\{Cell_r \in RC | Cell_r \in \underline{C}_{GCO}(X)\} \cup \{Cell_c D_i \in CC | RS(Cell_c) \in \underline{C}_{GCO}(X)\})$$

$$\underline{C}_{DEO}(X) = \cup (\{Cell_r \in RC | Cell_r \in \underline{C}_{GCO}(X)\} \cup \{Cell_c \in CC | RS(Cell_c) \cap \underline{C}_{GCO}(X) \neq \emptyset\})$$

$$\overline{C}_{DAO}(X) = \cup (\{Cell_r \in RC | Cell_r \in \overline{C}_{GCO}(X)\} \cup \{Cell_c \in CC | RS(Cell_c) \in \overline{C}_{GCO}(X)\})$$

$$\overline{C}_{DEO}(X) = \cup (\{Cell_r \in RC | Cell_r \in \overline{C}_{GC0}(X)\} \cup \{Cell_c \in CC | RS(Cell_c) \cap \overline{C}_{GCO}(X) \neq \emptyset\})$$

$$\overline{C}_{GCO}(X) = \{Cell_r \in RC | Cell_r \cap X \neq \emptyset\}$$

$$\underline{C}_{GCO}(X) = \{Cell_r \in RC | Cell_r \in X\}$$

Wu et al[14] implemented the granularization of the semi-monolayer covering rough set approximation operator, which provides a new idea of update operation for dynamic update in distributed computing, which can be decomposed into "object to granule" phase update and "granule to approximate set" phase update. To this end, we rewrite Definition 3 using the information interpretation of the unit set to create an independent expression for the "grain to approximate set".

Definition 4 Let $S = (U, A \cup d, V, f)$ be a set-valued decision information system and the approximation set consisting of $\overrightarrow{Cell}$ is defined as:

$$\overline{C}_{GC0}{}'(X) = \{\overrightarrow{Cell_r} | Cell_r \cap X \neq \emptyset\}$$

$$\underline{C}_{GC0}{}'(X) = \{\overrightarrow{Cell_r} | Cell_r \subseteq X\}$$

$$\underline{C}_{DA0}{}'(X) = \{\overrightarrow{Cell_r} | \overrightarrow{Cell_r} \in \underline{C}_{GC0}{}'(X)\} \cup \{\overrightarrow{Cell_c} | RS(Cell_c) \subseteq \underline{C}_{GC0}{}'(X)\}$$

$$\underline{C}_{DE0}{}'(X) = \{\overrightarrow{Cell_r} | \overrightarrow{Cell_r} \in \underline{C}_{GC0}{}'(X)\} \cup \{\overrightarrow{Cell_c} | RS(Cell_c) \cap \underline{C}_{GC0}{}'(X) \neq \emptyset\}$$

$$\overline{C}_{DA0}{}'(X) = \left\{\overrightarrow{Cell_r} | \overrightarrow{Cell_r} \in \overline{C}_{GC0}{}'(X)\right\} \cup \{\overrightarrow{Cell_c} | RS(Cell_c) \subseteq \overline{C}_{GC0}{}'(X)\}$$

$$\overline{C}_{DE0}{}'(X) = \left\{\overrightarrow{Cell_r} | \overrightarrow{Cell_r} \in \overline{C}_{GC0}{}'(X)\right\} \cup \{\overrightarrow{Cell_c} | RS(Cell_c) \cap \overline{C}_{GC0}{}'(X) \neq \emptyset\}$$

Table 1: Set-value decision table

Object	Language Quality	Originality	Review Conclusion
1	G	G	Ac
2	G	G	Ac
3	G	G	Ac
4	GP	G	Ac
5	GP	G	Ac
6	P	G	Ac
7	P	G	Ac
8	P	G	Re
9	GP	P	Re
10	G	P	Re
11	G	GP	Ac
12	P	GP	Re
13	P	GP	Re
14	G	P	Re
15	G	P	Re
16	GP	P	Re
17	P	P	Re

Example 1: A journal editor invited two reviewers to evaluate the seventeen articles. Two reviewers

ISSN 2616-5775 Vol. 5, Issue 14: 11-21, DOI: 10.25236/AJCIS.2022.051402

evaluated each article from two perspectives: language quality and originality. The results of each test are defined as either good or bad. Two reviewers may give different conclusions about the same article, which will lead to inconsistent results. If the results of the two review experts are combined, the journal will obtain a set value information table, as shown in Table 1.

In Table 1, $S=(U,A\cup d,V,f))$is A set value decision information table,A ={language quality, originality}, d={Ac, Re},f={G, P, Re, Ac, where G, P, re, and AC represent good, bad, reject and accept, and GP is set value, representing inconsistent review conclusions of experts. For the sake of narration, let "G" =1, "P" =2, "GP" =3.

$$\overrightarrow{Cell_1}=\overrightarrow{\{x_1,x_2,x_3\}}=<1,1> \quad \overrightarrow{Cell_2}=\overrightarrow{\{x_6,x_7,x_8\}}=<2,1>$$

$$\overrightarrow{Cell_3}=\overrightarrow{\{x_{14},x_{15}\}}=<1,2> \; \overrightarrow{Cell_4}=\overrightarrow{\{x_{17}\}}=<2,2>$$

$$\overrightarrow{Cell_5}=\overrightarrow{\{x_4,x_5\}}=<3,1> \quad \overrightarrow{Cell_6}=\overrightarrow{\{x_9,x_{10}\}}=<1,3>$$

$$\overrightarrow{Cell_7}=\overrightarrow{\{x_{11}\}}=<3,3> \quad \overrightarrow{Cell_8}=\overrightarrow{\{x_{12},x_{13}\}}=<2,3>$$

$$\overrightarrow{Cell_9}=\overrightarrow{\{x_{16}\}}=<3,2>$$

Explain whether can be seen from the above, information unit contains the set value, can be divided into $\overrightarrow{CC}$will explain information units, or $\overrightarrow{RC}$:

$$\overrightarrow{RC}=\left\{\overrightarrow{Cell_1},\overrightarrow{Cell_2},\overrightarrow{Cell_3},\overrightarrow{Cell_4}\right\}=\{<1,1>,<2,1>,<1,2>,<2,2>\}$$

$$\overrightarrow{CC}=\left\{\overrightarrow{Cell_5},\overrightarrow{Cell_6},\overrightarrow{Cell_7},\overrightarrow{Cell_8},\overrightarrow{Cell_9}\right\} \;=\{<3,1>,<1,3>,<3,3>,<2,3>,<3,2>\}$$

Decision set:

$$\mathrm{U/d}=\{\mathrm{Ac}=\{x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_{11}\}\ \mathrm{Re}=\{x_8,x_9,x_{10},x_{12},x_{13},x_{14},x_{15},x_{16},x_{17}\}\}$$

The decision set information is interpreted as:

$$\{<1,1>,<3,1>,<2,1>,<3,3>\}\subseteq\overrightarrow{Ac}$$

By definition 1, 4, according to the decision set and correlation $\overrightarrow{\mathrm{RC}}$as well$\overrightarrow{\mathrm{Cell_r}}\subseteq\overrightarrow{\mathrm{Ac}}$, $\overrightarrow{\mathrm{Cell_r}}\cap\overrightarrow{\mathrm{Ac}}\neq\emptyset$, available$\overline{\mathrm{C}}_{\mathrm{GC0}}{}'(\mathrm{Ac}),\underline{\mathrm{C}}_{\mathrm{GC0}}{}'(\mathrm{Ac})$ is as follows:

$$\overline{C}_{GC0}{}'(Ac)=\{<1,1>,<2,1>\}\ \underline{C}_{GC0}{}'(Ac)=\{<1,1>\}$$

$\mathrm{RS}(\mathrm{Cell_c})$ can be solved in terms of the relationship between the reliable and disputed elements, where RS is the current CC dependent RC:

$$RS(Cell_5)=\left\{\overrightarrow{Cell_1},\overrightarrow{Cell_2}\right\}=\{<1,1>,<2,1>\}$$

$$RS(Cell_6)=\left\{\overrightarrow{Cell_1},\overrightarrow{Cell_3}\right\}=\{<1,1>,<1,2>\}$$

$$RS(Cell_7)=\left\{\overrightarrow{Cell_1},\overrightarrow{Cell_2},\overrightarrow{Cell_3},\overrightarrow{Cell_4}\right\}=\{<1,1>,<2,1>,<1,2>,<2,2>\}$$

$$RS(Cell_8)=\left\{\overrightarrow{Cell_2},\overrightarrow{Cell_4}\right\}=\{<2,1>,<2,2>\}$$

$$RS(Cell_9)=\left\{\overrightarrow{Cell_3},\overrightarrow{Cell_4}\right\}=\{<1,2>,<2,2>\}$$

According to the above definition 2, 4, the approximate set$\underline{\mathrm{C}}_{\mathrm{DA0}}{}'(\mathrm{Ac})$, $\underline{\mathrm{C}}_{\mathrm{DE0}}{}'(\mathrm{Ac})$, $\underline{\mathrm{C}}_{\mathrm{DE0}}{}'(\mathrm{Ac})$, $\overline{\mathrm{C}}_{\mathrm{DA0}}{}'(\mathrm{Ac})$, $\overline{\mathrm{C}}_{\mathrm{DE0}}{}'(\mathrm{Ac})$ as follows:

$$\underline{C}_{DA0}{}'(Ac)=\underline{C}_{GC0}{}'(Ac),\ \underline{C}_{DE0}{}'(Ac)=\{<3,1>,<1,3>,<3,3>\}\cup\underline{C}_{GC0}{}'(Ac)$$

$$\overline{C}_{DA0}{}'(Ac)=\{<3,1>\}\cup\overline{C}_{GC0}{}'(Ac),\overline{C}_{DE0}{}'(Ac)=\{<3,1>,<1,3>,<3,3>,<2,3>\}\cup\overline{C}_{GC0}{}'(Ac)$$

3. Incremental update the semi-monolayer covering rough set approximation

This section provides an overall idea for updating the $\mathrm{DA0}',\mathrm{DE0}'$of the quasi-single-layer covering rough sets when adding object sets.

Definition 5. Let S^{t_0} be the set-valued decision information system at moment t_0, $S^{\Delta t}$be the new set-valued decision information system at moment Δt, and $S^{t_o+\Delta t} = S^{t_o} \cup S^{\Delta t}$.

(1) For $Cell^{\Delta t}$, if $\exists Cell^{t_o} \in CELL^{t_o}$, $\overrightarrow{Cell^{t_o}} = \overrightarrow{Cell^{\Delta t}}$, then the cell $Cell^{\Delta t}$ is said to extend $Cell^{t_o}$, when $Cell^{t_o+\Delta t} = Cell^{t_o} \cup Cell^{\Delta t}$.

(2) For $X^{\Delta t} \in U^{\Delta t}/D$, if $\exists X^{t_o} \in U^{t_o}/D$, $\overrightarrow{X^{t_o}} = \overrightarrow{X^{\Delta t}}$, then the decision set $X^{\Delta t}$ is said to extend X^{t_o}, at which time $X^{t_o+\Delta t} = X^{t_o} \cup X^{\Delta t}$.

(3) For $Cell^{\Delta t}$, if $\forall Cell^{t_o} \in CELL^{t_o}$, $\overrightarrow{Cell^{t_o}} \neq \overrightarrow{Cell^{\Delta t}}$, then the cell $Cell^{\Delta t}$ is said to be added, at which time $Cell^{t_o+\Delta t} = Cell^{\Delta t}$.

According to the previous description, the update of the approximation sets $DA0'$ and $DE0'$ can be divided into two steps.

Step 1: Update the information unit set to update the mapping relationship between "object and unit set" on the one hand; on the other hand, update the list of unit sets to provide the basis for the subsequent grain-based approximation set update.

Step 2: Based on the updated information unit set, the impact on the semi-monolayer covering rough set approximation set is discussed as follows.

(1) The effect of the added reliable units on the $GC0'$ approximation set.

(2) The effect of the added controversial units on $DA0', DE0'$ approximation sets.

(3) The update of the dependency matrix of the new reliable and controversial units.

Information Unit Set Update

Clustering the elements in $S^{\Delta t}$ by information interpretation, the category $Cell^{\Delta t}$ to which $x^{\Delta t}$ belongs, forms $CELL^{\Delta t}$.

Whether an interpretation different from $CELL^{t_o}$ appears in $CELL^{\Delta t}$; if it does, extend $CELL^{t_o}$; otherwise, maintain the same.

First determine whether the $\overrightarrow{Cell}$ corresponding to the current x already exists; if it does, no new information unit is generated; if not, a new information unit is generated, and the update of the mapping relationship is completed.

When a new object is added with $x^{\Delta t} \in Cell^{\Delta t}$, it is necessary to consider the impact of $\overrightarrow{Cell^{\Delta t}}$ on the original system. For this purpose, we establish update strategy 1:

Update Strategy 1. $S^{t_o} = (U^{t_o}, A \cup D, V, f)$ is the set-valued decision information system at moment t_0. $S^{\Delta t}$ is the newly added set-valued decision system at moment Δt and $CELL^{\Delta t}$ is the set of information units of $S^{\Delta t}$ such that $Cell^{\Delta t} \in CELL^{\Delta t}$.

(1) If there exists $Cell^{t_o} \in CELL^{t_o}$, and $Cell^{\Delta t}$ is an extension of $Cell^{t_o}$, then $Cell^{\Delta t}$ need not be added to $CELL^{t_o+\Delta t}$.

(2) If any $Cell^{t_o} \in CELL^{t_o}$ and $Cell^{\Delta t}$ is an added information cell, then $Cell^{\Delta t} \in CELL^{t_o+\Delta t}$.

4. Approximate centralized reliable unit update

4.1. $C_{GC0}'(X)$ under approximate dynamic change update

Theorem 1. If $S^{t_o} = (U^{t_o}, A \cup D, V, f)$ is the set-valued decision information system at moment t_0, ${Cell_r}^{t_0}$ is the reliable information unit in S^{t_o}, and $\overrightarrow{RC^{t_o}}$ is the set of reliable information unit interpretation at moment t_0. $S^{\Delta t}$ is the incremental set-valued at moment Δt Decision information system, $CELL^{\Delta t}$ is the full set of units of $S^{\Delta t}$, ${Cell_r}^{\Delta t}$ is the reliable information unit in $S^{\Delta t}$, if $X^{\Delta t}$ extends X^{t_o}, the approximate set under $GC0'$ is.

(1) If$\exists {Cell_r}^{t_o} \in \underline{C}_{GC0}'(X^{t_o})$,${Cell_r}^{\Delta t}$extends ${Cell_r}^{t_o}$,and${Cell_r}^{\Delta t} \subseteq X^{\Delta t}$,then$\overrightarrow{{Cell_r}^{t_o+\Delta t}} \in \underline{C}_{GC0}'(X^{t_o+\Delta t})$, that is.$\underline{C}_{GC0}'(X^{t_o})$ does not change.

ISSN 2616-5775 Vol. 5, Issue 14: 11-21, DOI: 10.25236/AJCIS.2022.051402

(2) If $\exists \mathrm{Cell_r}^{t_o} \in \underline{C}_{GC0}'(X^{t_o})$, $\mathrm{Cell_r}^{\Delta t}$ extends $\mathrm{Cell_r}^{t_o}$ but $\mathrm{Cell_r}^{\Delta t} \nsubseteq X^{\Delta t}$, then $\overrightarrow{\mathrm{Cell_r}^{t_o+\Delta t}} \notin \underline{C}_{GC0}'(X^{t_o+\Delta t})$, that is $\underline{C}_{GC0}'(X^{t_o+\Delta t}) = \underline{C}_{GC0}'(X^{t_o}) - \{\overrightarrow{\mathrm{Cell_r}^{t_o}}\}$.

(3) If $\mathrm{Cell_r}^{\Delta t}$ is a new information unit and $\mathrm{Cell_r}^{\Delta t} \subseteq X^{\Delta t}$, then $\overrightarrow{\mathrm{Cell_r}^{\Delta t}} \in \underline{C}_{GC0}'(X^{t_o+\Delta t})$, i.e. $\underline{C}_{GC0}'(X^{t_o+\Delta t}) = \underline{C}_{GC0}'(X^{t_o}) \cup \{\overrightarrow{\mathrm{Cell_r}^{\Delta t}}\}$.

Proof: (1) Because $\mathrm{Cell_r}^{t_o} \in \underline{C}_{GC0}'(X^{t_o})$, according to Definition 4, so $\mathrm{Cell_r}^{t_o} \subseteq X^{t_o}$. Because $\mathrm{Cell_r}^{\Delta t} \subseteq X^{\Delta t}$, so $\mathrm{Cell_r}^{t_o} \cup \mathrm{Cell_r}^{\Delta t} \subseteq X^{t_o} \cup X^{\Delta t}$. And because $\mathrm{Cell_r}^{\Delta t}$extends to $\mathrm{Cell_r}^{t_o}$, by definition 5, $\mathrm{Cell}^{t_o+\Delta t} = \mathrm{Cell}^{t_o} \cup \mathrm{Cell}^{\Delta t}$, $X^{t_o+\Delta t} = X^{t_o} \cup X^{\Delta t}$, and we can get $\overrightarrow{\mathrm{Cell}^{t_o+\Delta t}} = \overrightarrow{\mathrm{Cell}^{t_o}} = \overrightarrow{\mathrm{Cell}^{\Delta t}}$, then $\overrightarrow{\mathrm{Cell}^{t_o+\Delta t}} \in \underline{C}_{GC0}'(X^{t_o+\Delta t})$.

(2) Because $\mathrm{Cell_r}^{t_o} \in \underline{C}_{GC0}'(X^{t_o})$, according to Definition 4, so $\mathrm{Cell_r}^{t_o} \subseteq X^{t_o}$. And because $\mathrm{Cell_r}^{\Delta t} \nsubseteq X^{\Delta t}$, so $\mathrm{Cell_r}^{\Delta t} \cap X^{\Delta t} \neq \emptyset$, that is, $\mathrm{Cell_r}^{t_o} \cup \mathrm{Cell_r}^{\Delta t} \nsubseteq X^{t_o} \cup X^{\Delta t}$. And because $\mathrm{Cell_r}^{\Delta t}$ extends to $\mathrm{Cell_r}^{t_o}$, it follows from Definition 5 that $\mathrm{Cell}^{t_o+\Delta t} = \mathrm{Cell}^{t_o} \cup \mathrm{Cell}^{\Delta t}$, $X^{t_o+\Delta t} = X^{t_o} \cup X^{\Delta t}$, and $\overrightarrow{\mathrm{Cell}^{t_o+\Delta t}} \notin \underline{C}_{GC0}'(X^{t_o+\Delta t})$

(3) It is known that $\mathrm{Cell_r}^{\Delta t}$ is a newly generated reliable cell, $\forall \overrightarrow{\mathrm{Cell_r}^{t_o}} \neq \overrightarrow{\mathrm{Cell_r}^{\Delta t}}$. Since $\mathrm{Cell_r}^{\Delta t} \subseteq X^{\Delta t}$, so $\mathrm{Cell_r}^{\Delta t} \subseteq X^{t_o} \cup X^{\Delta t}$. By Definition 5, $X^{t_o+\Delta t} = X^{t_o} \cup X^{\Delta t}$, so $\mathrm{Cell_r}^{\Delta t} \subseteq X^{t_o+\Delta t}$, and by Definition 4, $\overrightarrow{\mathrm{Cell_r}^{\Delta t}} \in \underline{C}_{GC0}'(X^{t_o+\Delta t})$.

Under the premise of completing the mapping relationship update, it can be seen from Definition 4 that the factors affecting the approximation set are $\underline{C}_{GC0}'(X)$, $\overline{C}_{GC0}'(X)$, $\overrightarrow{\mathrm{Cell_r}} \in \overrightarrow{RC}, \overrightarrow{\mathrm{Cell_c}} \in \overrightarrow{CC}$, where $\underline{C}_{GC0}'(X)$, $\overline{C}_{GC0}'(X)$ is not only affected by $\overrightarrow{\mathrm{Cell}^{\Delta t}}$, but also related to $\overrightarrow{X_j^{\Delta t}}$.

When a new object is added$x^{\Delta t} \in U^{\Delta t}$, $\overrightarrow{x^{\Delta t}} = \overrightarrow{\mathrm{Cell}^{\Delta t}} + \overrightarrow{X_j^{\Delta t}}$, and if Cell is a reliable element, it needs to consider $\overrightarrow{\mathrm{Cell}^{\Delta t}} + \overrightarrow{X_j^{\Delta t}}$ on the original system $\underline{C}_{GC0}'(X)$ impact. For this purpose, we establish update strategy 2:

Update Strategy 2. $S^{t_o} = (U^{t_o}, A \cup D, V, f)$ is the set-valued decision information system at moment t_0, $U^{t_o}/D = \{X_i^{t_o} | i = (1,2, \cdots m)\}$, $S^{\Delta t} = (U^{\Delta t}, A \cup D, V, f)$ is the newly added set-valued decision information system at moment Δt, $U^{\Delta t}/D = \{X_j^{\Delta t} | j = (1,2, \cdots n)\}$, $\mathrm{Cell_r}^{\Delta t}$ is the reliable information unit of the added object set, $S^{t_o+\Delta t}$ is the set-valued decision information system at the moment of $t_0 + \Delta t$, such that $\overrightarrow{\mathrm{Cell_r}^{t_o}} \in \underline{C}_{GC0}'(X_i^{t_o})$, $\mathrm{Cell_r}^{t_o} \subseteq X_i^{t_o}$, $X_i^{t_o} \in U^{t_o}/D$, $X_j^{\Delta t} \in U^{\Delta t}/D$, for $X_j^{\Delta t}$ extends $X_i^{t_o}$:

(1) If $\mathrm{Cell_r}^{\Delta t} \subseteq X_j^{\Delta t}$, $\mathrm{Cell_r}^{\Delta t}$ extends $\mathrm{Cell_r}^{t_o}$, and $\overrightarrow{\mathrm{Cell_r}^{\Delta t}} \in \underline{C}_{GC0}'(X_i^{t_o})$, then$\overrightarrow{\mathrm{Cell_r}^{\Delta t}}$is not added to the approximation set under $GC0'$.

(2) If $\mathrm{Cell_r}^{\Delta t} \nsubseteq X_j^{\Delta t}$, $\mathrm{Cell_r}^{\Delta t}$extends $\mathrm{Cell_r}^{t_o}$, and $\overrightarrow{\mathrm{Cell_r}^{\Delta t}} \in \underline{C}_{GC0}'(X_i^{t_o})$, then $\overrightarrow{\mathrm{Cell_r}^{\Delta t}}$does not belong to $GC0'$ under the approximation set.

(3) If $\mathrm{Cell_r}^{\Delta t} \subseteq X^{\Delta t}$ and $\mathrm{Cell_r}^{\Delta t}$ does not extend $\mathrm{Cell_r}^{t_o}$, then $\overrightarrow{\mathrm{Cell_r}^{\Delta t}}$ belongs to the approximation set under $GC0'$.

4.2. $\overline{C}_{GCO}'(X)$ on approximate dynamic change update

Theorem 2. If $S^{t_o} = (U^{t_o}, A \cup D, V, f)$ is the set-valued decision information system at moment t_0, $\mathrm{Cell_r}^{t_o}$ is the reliable information unit in S^{t_o}, and $\overrightarrow{RC^{t_o}}$is the reliable unit information interpretation set at moment t_0. $S^{\Delta t}$ is the incremental set-valued at moment Δt decision information system, $CELL^{\Delta t}$ is the full set of units in $S^{\Delta t}$, $\mathrm{Cell_r}^{\Delta t}$ is the reliable information unit in $S^{\Delta t}$, if $X^{\Delta t}$ extends X^{t_o}, then $GC0'$ on the approximation set is:

(1) If $\exists \mathrm{Cell_r}^{t_o} \in \overline{C}_{GC0}'(X^{t_o})$ and $\mathrm{Cell_r}^{\Delta t}$ extends $\mathrm{Cell_r}^{t_o}$, then $\overrightarrow{\mathrm{Cell_r}^{t_o+\Delta t}} \in \overline{C}_{GC0}'(X^{t_o+\Delta t})$. that $\overline{C}_{GC0}'(X^{t_o})$ does not change.

ISSN 2616-5775 Vol. 5, Issue 14: 11-21, DOI: 10.25236/AJCIS.2022.051402

(2) If ${Cell_r}^{\Delta t}$ extends ${Cell_r}^{t_0}$ and ${Cell_r}^{\Delta t} \cap X^{\Delta t} \neq \emptyset$, then $\overrightarrow{{Cell_r}^{t_0+\Delta t}} \in \overline{C}_{GC0}{}'(X^{t_0+\Delta t})$, i.e. $\overrightarrow{{Cell_r}^{\Delta t}}$is incorporated into $\overline{C}_{GC0}{}'(X^{t_0})$.

(3) If ${Cell_r}^{\Delta t}$ is added and ${Cell_r}^{\Delta t} \cap X^{\Delta t} \neq \emptyset$, then$\overrightarrow{{Cell_r}^{\Delta t}} \in \overline{C}_{GC0}{}'(X^{t_0+\Delta t})$, i.e., $\overrightarrow{{Cell_r}^{\Delta t}}$ is merged into $\overline{C}_{GC0}{}'(X^{t_0})$.

Proof: (1) Since ${Cell_r}^{t_0} \in \overline{C}_{GC0}{}'(X^{t_0})$ by Definition 5, ${Cell_r}^{t_0} \cap X^{t_0} \neq \emptyset$. And because ${Cell_r}^{\Delta t}$ extends ${Cell_r}^{t_0}$, by Definition 5, $Cell^{t_0+\Delta t} = Cell^{t_0} \cup Cell^{\Delta t}$ and $\overrightarrow{Cell^{t_0+\Delta t}} = \overrightarrow{Cell^{t_0}} = \overrightarrow{Cell^{\Delta t}}$, then $\overrightarrow{Cell^{t_0+\Delta t}} \in \overline{C}_{GC0}{}'(X^{t_0+\Delta t})$.

(2) Since ${Cell_r}^{t_0} \in \overline{C}_{GC0}{}'(X^{t_0})$, by Definition 4, ${Cell_r}^{t_0} \cap X^{t_0} \neq \emptyset$. And because ${Cell_r}^{\Delta t} \cap X^{\Delta t} \neq \emptyset$, so$(Cell^{t_0} \cup Cell^{\Delta t}) \cap (X^{t_0} \cup X^{\Delta t}) \neq \emptyset$. From Definition 5, $Cell^{t_0+\Delta t} = Cell^{t_0} \cup Cell^{\Delta t}$, $X^{t_0+\Delta t} = X^{t_0} \cup X^{\Delta t}$, and $\overrightarrow{Cell^{t_0+\Delta t}} = \overrightarrow{Cell^{t_0}} = \overrightarrow{Cell^{\Delta t}}$, then we can get $\overrightarrow{Cell^{t_0+\Delta t}} \in \overline{C}_{GC0}{}'(X^{t_0+\Delta t})$.

(3) It is known that ${Cell_r}^{\Delta t}$ is added, then a new reliable cell ${Cell_r}^{\Delta t}$ is generated at the moment of Δt, $\forall \overrightarrow{{Cell_r}^{t_0}} \neq \overrightarrow{{Cell_r}^{\Delta t}}$.Since ${Cell_r}^{\Delta t} \cap X^{\Delta t} \neq \emptyset$,${Cell_r}^{\Delta t} \cap (X^{t_0} \cup X^{\Delta t}) \neq \emptyset$.From Definition 5, $X^{t_0+\Delta t} = X^{t_0} \cup X^{\Delta t}$, so ${Cell_r}^{\Delta t} \cap X^{t_0+\Delta t} \neq \emptyset$, then $\overrightarrow{{Cell_r}^{\Delta t}} \in \overline{C}_{GC0}{}'(X^{t_0+\Delta t})$.

When a new object is added $x^{\Delta t} \in U^{\Delta t}$, $\overrightarrow{x^{\Delta t}} = \overrightarrow{Cell^{\Delta t}} + \overrightarrow{{X_j}^{\Delta t}}$, if Cell is a reliable element, we need to consider$\overrightarrow{{Cell_r}^{\Delta t}} + \overrightarrow{{X_j}^{\Delta t}}$ on the original system $\overline{C}_{GC0}{}'(X_i)$ impact. To this end, we establish update strategy 3:

Update strategy 3. $S^{t_0} = (U^{t_0}, A \cup D, V, f)$ is the set-valued decision information system at moment t_0, $U^{t_0}/D = \{{X_i}^{t_0} | i = (1,2,\cdots m)\}$, $S^{\Delta t} = (U^{\Delta t}, A \cup D, V, f)$ is the set-valued decision information system added at moment Δt, $U^{\Delta t}/D = \{{X_j}^{\Delta t} | j = (1,2,\cdots n)\}$, ${Cell_r}^{\Delta t}$ is the reliable information unit of the added object set, and $S^{t_0+\Delta t}$ is the set-valued decision information system at the moment of $t_0 + \Delta t$, such that$\overrightarrow{{Cell_r}^{t_0}} \in \overline{C}_{GC0}{}'({X_i}^{t_0})$, ${Cell_r}^{t_0} \subseteq {X_i}^{t_0}$, ${X_i}^{t_0} \in U^{t_0}/D$, ${X_j}^{\Delta t} \in U^{\Delta t}/D$, for ${X_j}^{\Delta t}$ extends ${X_i}^{t_0}$:

(1) If ${Cell_r}^{\Delta t} \subseteq {X_j}^{\Delta t}$, ${Cell_r}^{\Delta t}$ extends ${Cell_r}^{t_0}$ and $\overrightarrow{{Cell_r}^{\Delta t}} \in \overline{C}_{GC0}{}'({X_i}^{t_0})$, then$\overrightarrow{{Cell_r}^{\Delta t}}$ is not added to the approximation on $GC0'$.

(2) If ${Cell_r}^{\Delta t} \cap X^{\Delta t} \neq \emptyset$ and ${Cell_r}^{\Delta t}$ extends ${Cell_r}^{t_0}$, then $\overrightarrow{{Cell_r}^{\Delta t}}$ belongs to the $GC0'$ on approximation.

(3) If ${Cell_r}^{\Delta t} \cap X^{\Delta t} \neq \emptyset$ or ${Cell_r}^{\Delta t} \subseteq {X_j}^{\Delta t}$, but ${Cell_r}^{\Delta t}$ does not extend ${Cell_r}^{t_0}$, then $\overrightarrow{{Cell_r}^{\Delta t}}$ belongs to $GC0'$on approximation.

4.3. RSM Changes Update

$S^{t_0} = (U^{t_0}, A \cup D, V, f)$ is the set-valued decision information system at moment t_0, $\overrightarrow{RC^{t_0}}$ is the set of reliable cell information interpretation at moment t_0 $\overrightarrow{RC^{t_0}} = \{\overrightarrow{{Cell_{r1}}^{t_0}}, \overrightarrow{{Cell_{r2}}^{t_0}}, \cdots, \overrightarrow{{Cell_{rm}}^{t_0}}\}$, $\overrightarrow{CC^{t_0}}$ is the set of contention cell information interpretation in the system at moment t_0, $\overrightarrow{CC^{t_0}} = \{\overrightarrow{{Cell_{c1}}^{t_0}}, \overrightarrow{{Cell_{c2}}^{t_0}}, \cdots, \overrightarrow{{Cell_{cm}}^{t_0}}\}$; where RSM^{t_0} is the relationship matrix of all disputed units and reliable units at the moment of t_0. $S^{\Delta t} = (U^{\Delta t}, A \cup D, V, f)$ is the set-valued decision information system added at Δt, $\overrightarrow{RC^{\Delta t}}$ is the set of reliable cell information explanation added at Δt, $\overrightarrow{RC^{\Delta t}} = \{\overrightarrow{{Cell_{r(m+1)}}^{\Delta t}}, \overrightarrow{{Cell_{r(m+2)}}^{\Delta t}}, \cdots, \overrightarrow{{Cell_{rM}}^{\Delta t}}\}$, $\overrightarrow{CC^{\Delta t}}$ is the set of new reliable cell information interpretation at Δt moment, $\overrightarrow{CC^{\Delta t}}$is the set of new The set of disputed cell information interpretation, $\overrightarrow{CC^{\Delta t}} = \{\overrightarrow{{Cell_{c(m+1)}}^{\Delta t}}, \overrightarrow{{Cell_{c(m+2)}}^{\Delta t}}, \cdots, \overrightarrow{{Cell_{cN}}^{\Delta t}}\}$.

If the added objects have set values and single values, the system reliable cells and disputed cells will be added at the same time, i.e. the RSM matrix needs to add ${RSM_L}^{t_0+\Delta t}$ part and ${RSM_C}^{t_0+\Delta t}$ part at the same time, ${RSM_{LC}}^{t_0+\Delta t}$ part will be added.

The three cases of $RSM^{t_0+\Delta t}$ correspond to Figure 1.

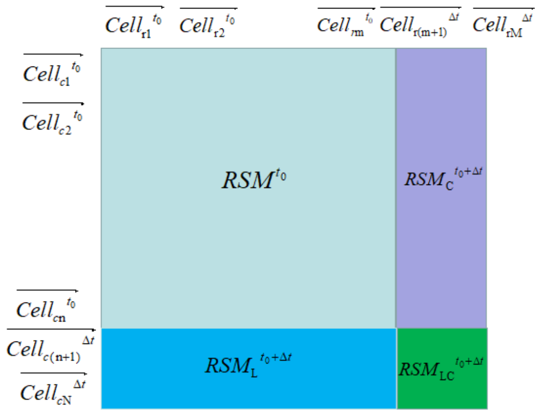


Figure 1: RSM changes at $t_0+\Delta t$ $(RSM^{t_0+\Delta t})$

For the three cases and one case where no information unit is added, we design an update strategy for the RSM.

Update strategy 4: When a new object is added $x^{\Delta t} \in U^{\Delta t}$, $\overrightarrow{x^{\Delta t}} = \overrightarrow{Cell^{\Delta t}} + \overrightarrow{X_j^{\Delta t}}$, if Cell is a reliable unit, what needs to be considered is $\overrightarrow{Cell_r^{\Delta t}} + \overrightarrow{X_j^{\Delta t}}$ on the original system $\overrightarrow{RC}$ impact, if the disputed element needs to consider $\overrightarrow{Cell_c}$ on $\overrightarrow{CC}$ impact of.

(1) $\overrightarrow{RC^{t_0}}$, $\overrightarrow{CC^{t_0}}$ does not change.

If $\forall\overrightarrow{Cell_r^{\Delta t}} \in \overrightarrow{RC^{t_0}}, \forall\overrightarrow{Cell_c^{\Delta t}} \in \overrightarrow{CC^{t_0}}$, i.e., the RSM matrix rows and columns do not change, then $RSM^{t_0+\Delta t}$=RSM^{t_0}.

(2) $\overrightarrow{RC^{t_0}}$ does not change and $\overrightarrow{CC^{t_0}}$ changes.

If $\forall\overrightarrow{Cell_r^{\Delta t}} \in \overrightarrow{RC^{t_0}}, \exists\overrightarrow{Cell_c^{\Delta t}} \notin \overrightarrow{CC^{t_0}}$, i.e., the RSM matrix rows appear to increase, then$RSM_L^{t_0+\Delta t}$=$\{(\overrightarrow{Cell_c^{\Delta t}}, RS^{t_0}(Cell_c^{\Delta t}))|\overrightarrow{Cell_c^{\Delta t}} \in \overrightarrow{CC^{t_0+\Delta t}}\}$, $RSM^{t_0+\Delta t} = RSM^{t_0} \cup RSM_L^{t_0+\Delta t}$.

(3) $\overrightarrow{RC^{t_0}}$ changes and $\overrightarrow{CC^{t_0}}$ does not change.

If $\exists\overrightarrow{Cell_r^{\Delta t}} \notin \overrightarrow{RC^{t_0}}, \forall\overrightarrow{Cell_c^{\Delta t}} \in \overrightarrow{CC^{t_0}}$, i.e., the RSM matrix column appears to increase, then $RSM_C^{t_0+\Delta t}$=$\{(\overrightarrow{Cell_c^{t_0}}, RS^{t_0+\Delta t}(Cell_c^{t_0}))|\ \overrightarrow{Cell_c^{t_0}} \in \overrightarrow{CC^{t_0}}\}$, $RSM^{t_0+\Delta t}$=$RSM_C^{t_0+\Delta t}$.

(4) $\overrightarrow{RC^{t_0}}$ changes and $\overrightarrow{CC^{t_0}}$ changes.

If $\exists\overrightarrow{Cell_r^{\Delta t}} \notin \overrightarrow{RC^{t_0}}, \exists\overrightarrow{Cell_c^{\Delta t}} \notin \overrightarrow{CC^{t_0}}$, i.e. RSM matrix rows and columns appear to increase simultaneously, then $RSM^{t_0+\Delta t} = RSM_L^{t_0+\Delta t} \cup RSM_C^{t_0+\Delta t} \cup \{(\overrightarrow{Cell_c^{\Delta t}}, RS^{\Delta t}(Cell_c^{\Delta t}))|\overrightarrow{Cell_c^{\Delta t}} \in \overrightarrow{CC^{\Delta t}}\}$.

5. Experimental

Experimental environment: The experiments are run on a master-slave structured Spark cluster, where the master node driver has 4 cores and 30G of memory, and the slave node has 16 cores and 64G of memory. The program is written in IntelliJ IDEA Community Edition 2020.3 x64, and the related software versions are: JDK 8, Spark 3.2.0, Hadoop 3.3.1, and Scala 2.12.15.

The experimental datasets are shown in Table 2, where the HAPMASS, HIGGS, and SUSY datasets are from UCI (University of California, Irvine machine learning database), and the other part of the data generation is generated from the weka software. Generator-related parameters: the generated data model is random forest, the total number of attributes is 21, the decision value is set to dichotomous, the generation type is continuous, the name of the generated dataset is Wekai0, and i denotes the number of samples (unit: million).

Specific experimental steps.

This experiment mainly compares the computation time of non-incremental and incremental algorithms on different datasets. We partition the dataset A into $A_1$and A_2, A_1 contains 50% of the randomly selected data volume as the base dataset, and A_2 contains the remaining 50% of the data and is equally partitioned into 10 equal parts Δa. The moment t_0 corresponds to the base dataset A_1, and each subsequent node corresponds to the respective running time of the static and incremental algorithms whenever 5% of the data volume is added. of running time. As in Figure 2, the horizontal coordinate in each subplot indicates the number of copies of Δa added, and the vertical coordinate is the model computation time, while the black dash shows the computation time of the static algorithm and the red dash represents the computation time of the incremental update algorithm. For example, in Figure 2(a), the moment Δt_1 indicates that a Δa is added to the data volume, at which time the solution of Δa by the incremental algorithm and the approximate set solution of $A_1 + \Delta a$ by the static algorithm constitute the first set of comparison experiments. The moment of Δt_3 indicates that the data volume is increased by 3 Δa, i.e., 15% of the data volume is added, and the solution of 3*Δa by the incremental algorithm and the approximate set solution of A_1+3*Δa by the static algorithm constitute the third set of comparison experiments.

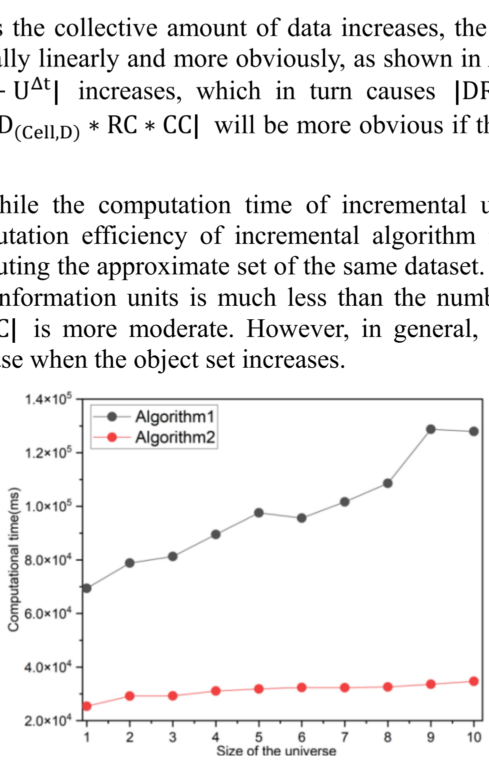
Table 2: Data sets

Dataset	Number of objects (million)	Number of attributes	Number of decisions	Source
HAPMASS	700	27	2	UCI
HIGGS	1100	28	2	UCI
SUSY	500	19	2	UCI
Weka20	2000	21	2	weka generator
Weka30	3000	21	2	weka generator
Weka40	4000	21	2	weka generator

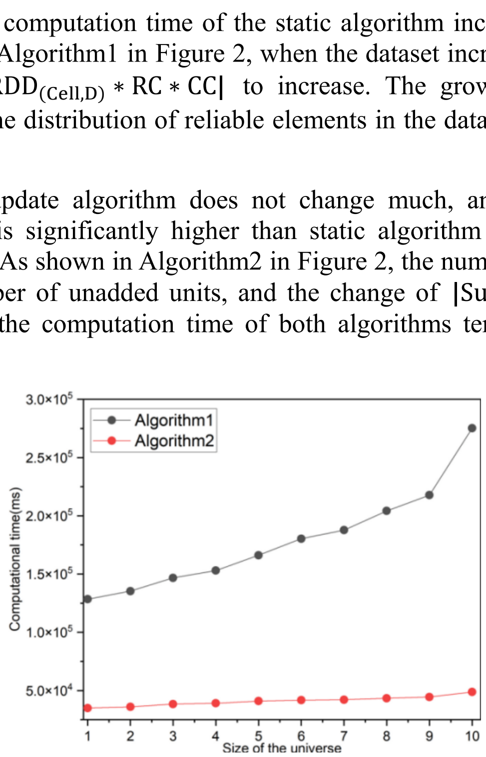
As can be seen in Figure 2, for each data set, the computation time of both the static algorithm (Algorithm1) and the incremental algorithm (Algorithm2) increases as the ratio of different data sets increases. The computation time of the incremental algorithm is much smaller than that of the static algorithm. In addition, the incremental algorithm has a more significant speedup for the last three datasets Weka20, Weka30 and Weka40. This is because the data are divided more evenly when performing the construction of the set-value decision system.

As the collective amount of data increases, the computation time of the static algorithm increases basically linearly and more obviously, as shown in Algorithm1 in Figure 2, when the dataset increases, $|U^{t_0} + U^{\Delta t}|$ increases, which in turn causes $|DRDD_{(Cell,D)} * RC * CC|$ to increase. The growth of $|DRDD_{(Cell,D)} * RC * CC|$ will be more obvious if the distribution of reliable elements in the data set is large.

While the computation time of incremental update algorithm does not change much, and the computation efficiency of incremental algorithm is significantly higher than static algorithm when computing the approximate set of the same dataset. As shown in Algorithm2 in Figure 2, the number of new information units is much less than the number of unadded units, and the change of $|SubCC * SubRC|$ is more moderate. However, in general, the computation time of both algorithms tends to increase when the object set increases.



(a) HAPMASS



(b) HIGGS

ISSN 2616-5775 Vol. 5, Issue 14: 11-21, DOI: 10.25236/AJCIS.2022.051402

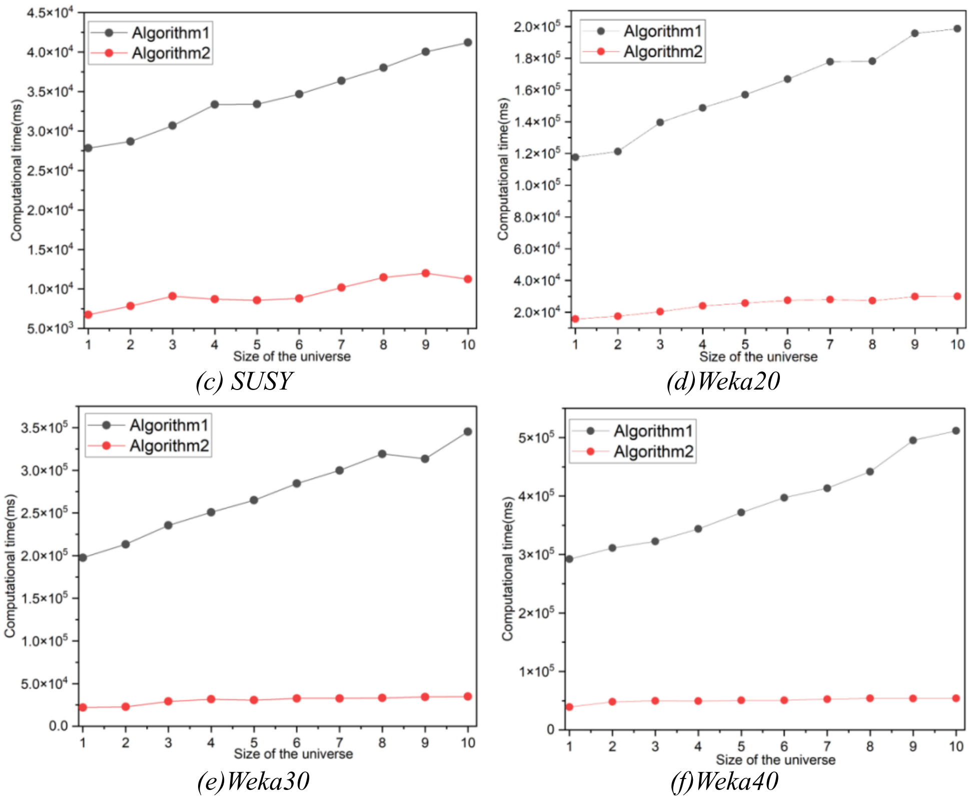


Figure 2: Comparison of Algorithm1 and Algorithm2 computation time when adding object sets

6. Conclusions

In the era of big data, we often encounter huge amount of data and the continuous increase of data volume. If our algorithm can dynamically update and compute the approximation set as the data increases, saving and maintaining the solved approximation set based on the previous updated data will save a lot of time and improve the computational efficiency.

In this paper, we use the semi-monolayer covering rough set to implement the dynamic update of the approximation set in Spark environment. When the object set is added, especially when it is increased massively, the reliable and disputed units in the set value information system will change, which will lead to the change of $\overline{C}_{GC0}{}'(X)$, $\underline{C}_{GC0}{}'(X)$ and. $RS(Cell_c)$ The correctness of the update strategy is demonstrated by case studies. Finally, a series of experiments are conducted on the UCI dataset and the weka generation dataset to record and analyze the respective running times of the incremental and static algorithms in the face of the equal increase of the dataset. The experimental results show that our proposed incremental algorithm is always more efficient than the static algorithm when the amount of data in the object set increases. Moreover, in the future, we can investigate how to update attribute set changes in Spark environment and the dynamic update of attribute value set changes, which is a common scenario for set-valued information systems.

References

[1] Zdzislaw Pawlak. Rough sets. International Journal of Computer and Information, 11:341–356, 1982.
[2] GAO Tianyu, WANG Qingrong, YANG Lei. Data Mining Model Based on Attribute Dependability Enhancement of Rough Set [J]. Computer Engineering and Applications, 2021, 57(3): 87-93.
[3] Yao Y. Three-way decision and granular computing [J]. International Journal of Approximate Reasoning, 2018, 103:107-123.
[4] Carlos Fernandez-Basso, M. Dolores Ruiz, and Maria J. Martin-Bautista. Spark solutions for discovering fuzzy association rules in big data. International Journal of Approximate Reasoning, 137:94–112, 2021.
[5] Chuan Luo, Tianrui Li, Hongmei Chen, and Lixia Lu. Fast algorithms for computing rough approximations in set-valued decision systems while updating criteria values.Information Sciences,

299:221–242, 2015.
[6] Jianfeng Xu, Duoqian Miao, Y uanjian Zhang, and Zhifei Zhang. A three-way decisions model with probabilistic rough sets for stream computing.International Journal of Approximate Reasoning, 88:1–22, 2017.
[7] Xiaojun Xie and Xiaolin Qin. A novel incremental attribute reduction approach for dynamic incomplete decision systems.International Journal of Approximate Reasoning, 93:443–462, 2018.
[8] Chengxiang Hu, Shixi Liu, and Xiaoling Huang. Dynamic updating approximations in multigranulation rough sets while refining or coarsening attribute values.Knowledge-Based Systems, 130:62–73, 2017. [9]
[9] Fanyu Bu, Zhikui Chen, Qingchen Zhang. A big data feature learning model supporting incremental updates [J]. Computer Engineering and Applications, 2015, 51(12):21-26.
[10] L. Dong and D. Chen. Incremental attribute reduction with rough set for dynamic datasets with simultaneously increasing samples and attributes.International Journal of Machine Learning and Cybernetics, 11(5), 2020.
[11] WU Yangyang, TANG Jianguo. Research Progress of Attribute Reduction Based on Rough Set in Context of Big Data[J]. Computer Engineering and Applications, 2019, 55(6): 31-38.
[12] Junbo Zhang, Tian rui Li, and Hongmei Chen. Composite rough sets for dynamic data mining. Information Sciences, 257:81–100, 2014.
[13] WU Z J, CHEN N, GAO Y. Semi-monolayer cover rough set: Concept, property and granular algorithm .Information Sciences[J], 2018, 456: 97-112.
[14] WU Z J, WANG H, CHEN N, et al. Semi-monolayer covering rough set on set-valued information systems and its efficient computation[J]. International Journal of Approximate Reasoning, 2021, 130: 83-106.
[15]WU Zhengjiang, YANG Tian, ZHENG Ailing, MEI Qiuyu, ZHANG Yaning. Study on Set-Valued Data Balancing Method by Semi-Monolayer Covering Rough Set[J]. Computer Engineering and Applications, 2022, 58(19): 166-173.