Incremental Attribute Reduction Based on Knowledge Granularity under Incomplete Data

Qiangqiang Zhong, Lei Wang*, Wen Yang and Chao Liu

1 School of Information Engineering, Nanchang Institute of Technology, Nanchang 330000, China
2 Jiangxi Provincial Key Laboratory of Water Information Cooperative Sensing and Intelligent Processing, Nanchang Institute of Technology, Nanchang 330000, China
Email: ezhoulei@163.com

Abstract. Traditional attribute reductions can’t be directly applied to miss attribute values and dynamic changes data. Therefore, how to dynamically update attribute reductions efficiently when incomplete data changes dynamically is great research value. For the change of the object, using matrix’s simple and clear characteristics, the paper focus on the incremental attribute reduction algorithm in matrix form of knowledge granularity. Firstly, the knowledge granularity is obtained by matrix which uses to measure the uncertainty of the system, then calculating the attribute importance, and using the attribute importance to devise a static algorithm, and then analyzing incremental matrix form when the object increases. The comparative experiments show that the incremental algorithm can effectively and efficiently update the result of attribute reduction as the object changes.

Keywords. Incomplete; knowledge granularity; attribute importance; object increase; incremental attribute reduction algorithm.

1. Introduction
Rough set is an effective implement to manage inaccurate and indeterminacy data, which put forward by Pawlak [1]. It has been extensively applied to machine learning [2] and rule extraction [3]. Rough set also has been used in attribute reduction, and attribute reduction are mainly based on positive domain [4], information entropy [5] and discernibility matrix [6]. In big data rea, the large amount of data and the dynamic changes of data requires more efficient algorithms. Non-incremental algorithms have been difficult to meet this need. Incremental method [7] is considered to be an effective way to deal with this kind of situation. Most of the current algorithms are used for complete data, but unable be directly used for incomplete data. On incomplete dynamic data processing, Shu et al. [9] introduced a simpler method to calculate tolerance class than classical idea, and proposed an incremental algorithm to calculate the attribute reduction of the dynamically increasing incomplete decision system. Through the model of dynamic rough set, Luo et al. [10] analysed the update strategy and update algorithm when objects are added and deleted. A method of designing incremental algorithm for incomplete data by binary discernibility matrix was proposed by Ding et al. [11]

This paper will define the relational matrix according to the tolerance relation for incomplete decision-making system, design the static attribute reduction algorithm through the knowledge granularity and attribute importance in matrix form, then analyse how the relational matrix changes when the number of objects increases. In this way, we can optimize the problem that static algorithm
needs to restart calculation every time when we add some data, and gain a more efficient dynamic algorithm.

2. Basic knowledge

**Definition 1.** [7] Let a quadruple \( S = \{U, C \cup D, V, f \} \) be a decision system, \( U \) represents a non-empty object set, \( C \) is the conditional attribute set, \( D \) represents decision attribute, \( V_a \) is a domain of \( a \), and \( f : U \times C \cup D \rightarrow V \) is an information function where \( \forall \alpha \in C \cup D \), \( x \in U \) with \( f(x, \alpha) \in V_a \). \( \exists \alpha \in C \), the data with missing attribute values (\( f(x, \alpha) = * \)) which called incomplete decision system and abbreviated as \( IS = \{U, C \cup D, V, f \} \).

**Definition 2.** [7] Assume that \( IS \) is an incomplete decision system. The definition of tolerance relation \( T(B) \) expressed as below:

\[
T(B) = \{(x, y) \in U \times U \mid s \in B, f(x, s) = f(y, s) = * \lor f(x, s) = f(y, s) = *\}
\]

(1)

**Definition 3.** [7] Let \( IS \) be an incomplete decision system. We can get tolerance class \( T_B(x) \) when \( B \subseteq C \):

\[
T_B(x) = \{y \mid (x, y) \in T(B), y \in U\}
\]

(2)

**Definition 4.** [7] Let \( IS \) be an incomplete decision system. On the basis of \( T(B) \) derives the relation matrix \( (M_{T_B}^{\tau})_{non} = (m_{ij})_{non} \).

\[
m_{ij} = \begin{cases}
1, & (x, y) \in T(B)\\
0, & (x, y) \not\in T(B)
\end{cases}, 1 \leq i, j \leq n
\]

(3)

For the convenience of later expression, \( (M_{T_B}^{\tau})_{non} \) is abbreviated as \( M_{T_B}^{\tau} \).

2.1. Knowledge Granulation in Incomplete Information Systems

**Definition 5.** [7] Let \( IS \) be an incomplete decision system, \( M_{T_B}^{\tau} = (m_{ij})_{non} \) is the corresponding relation matrix of the conditional attribute \( C \). Based on the relation matrix, the definition of \( GK_{U,C} (C) \) shows as below:

\[
GK_{U,C} (C) = \sum_{i=1}^{n} \sum_{j=1}^{m} m_{ij} = \frac{\text{sum}(M_{T_B}^{\tau})}{n^2} = \overline{M_{T_B}^{\tau}}
\]

(4)

\( \text{sum}(M_{T_B}^{\tau}) \) is the sum of all elements of \( M_{T_B}^{\tau} \), \( \overline{M_{T_B}^{\tau}} \) represents the arithmetic mean of \( M_{T_B}^{\tau} \).

**Definition 6.** [7] Let \( IS \) be an incomplete decision system. \( M_{T_C}^{\tau} \) and \( M_{T_D}^{\tau} \) are the corresponding tolerance relation matrix of attribute sets \( C \) and \( C \cup D \).

The definition of knowledge granularity \( GK_{U,C} (D\mid C) \) in matrix form is given as below:

\[
GK_{U,C} (D\mid C) = GK_{U,C} (C) - GK_{U,C} (C \cup D) = \overline{M_{T_C}^{\tau}} - \overline{M_{T_D}^{\tau}}
\]

(5)

**Definition 7.** Let \( IS \) be an incomplete decision system, \( M_{T_C}^{\tau}, M_{T_C}^{\tau_{\alpha}}, M_{T_C}^{\tau_{\cup\alpha}}, M_{T_D}^{\tau_{\cup\alpha}} \) respectively the relational matrix of their corresponding attribute sets, \( \forall a \in C \), the definition of the matrix representation of inner attribute importance as below:

\[
\text{Sig}_{U}^{inner}(a, C, D) = GK_{U,C} (D \mid (C - \{a\})) - GK_{U,C} (D \mid C) = M_{T_C}^{\tau_{\alpha}} - M_{T_D}^{\tau_{\cup\alpha}} - M_{T_C}^{\tau} + M_{T_D}^{\tau_{\cup\alpha}}
\]

(6)

**Definition 8.** Assume that \( IS \) is an incomplete decision system, \( \forall a \in (C - B) \), the definition of the external attribute importance as follows:
\[ \text{Sig}_U^{\text{inner}}(a, B, D) = \text{GK}_U(D \mid B) - \text{GK}_U(D \mid (B \cup \{a\})) = \bar{M}_{U}^{T} - M_{U}^{T(a,a)} - M_{U}^{T(a,a\mid a)} + M_{U}^{T(a,a\mid a\mid a)} \]  

**Definition 9.** Assume that \( IS \) is an incomplete decision system. The core of incomplete decision information system was defined as below:

\[ \text{Core}_C(D) = \{ a \in C \mid \text{Sig}_U^{\text{inner}}(a, C, D) > 0 \} \]

**Definition 10.** Let \( IS \) be an incomplete decision system. \( B \) can be called attribute reduction when both of the following facts are satisfied.

\( (1) \text{GK}_U(D \mid B) = \text{GK}_U(D \mid C) \quad (2) \forall a \in B, \text{GK}_U(D \mid (C - \{a\})) \neq \text{GK}_U(D \mid C) \)

2.2. Heuristic Attribute Reduction Algorithm

**Algorithm 1.** Heuristic attribute reduction method was on account of matrix form in incomplete decision system (HARIDS)

Enter: \( IS \)

Output: RED

(a) Initialized RED, then calculated \( \text{GK}_U(D \mid C) \) and \( \text{Sig}_U^{\text{inner}}(a, C, D) \)

(b) According Definition 9 to get \( \text{Core}_C(D) \), then \( \text{Core}_C(D) \) assigned to RED, calculated \( \text{GK}_U(D \mid RED) \)

(c) while \( \text{GK}_U(D \mid RED) = \text{GK}_U(D \mid C) \), execute (e), otherwise execute (d)

(d) when \( \text{GK}_U(D \mid RED) \neq \text{GK}_U(D \mid C) \), \( \forall a \in (C - \{RED\}) \), calculate \( \text{Sig}_U^{\text{inner}}(a, RED, D) \), then add the most significance of attribute to RED , then generate a new RED, until \( \text{GK}_U(D \mid RED) = \text{GK}_U(D \mid C) \)

(e) \( \forall a \in RED \), if \( \text{GK}_U(D \mid (RED - \{a\})) = \text{GK}_U(D \mid C) \), \( (RED - \{a\}) \rightarrow RED \), else retained RED.

(f) finally output RED, ends.

3. Incremental Attribute Reduction Algorithm under Object Change

3.1. Updating Mechanism of Relational Matrix after Adding Object Set

**Definition 11.** Let \( IS \) be an incomplete decision system. The newly added objects set is \( U_x = \{x_{i+1}, x_{i+2}, x_{i+3}, \cdots, x_{n+x}\} \). The relationship matrix \( (P_{U \cup U_x}^{T})_{\text{rer}} = (p_{ij})_{\text{rer}} \) of adding an new object set is shown as follows:

\[ p_{ij} = \begin{cases} 1, & (x_i, x_j) \not\in T_C, 1 \leq j \leq n, 1 \leq i \leq t \\ 0, & (x_i, x_j) \in T_C \end{cases} \]

**Definition 12.** Let \( IS \) be an incomplete decision system, the new object set relationship matrix is defined as \( (S_{U_x}^{T})_{\text{rer}} = (s_{ij})_{\text{rer}} : \)

\[ s_{ij} = \begin{cases} 1, & (x_i, x_j) \not\in T_C, 1 \leq j \leq t, 1 \leq i \leq t \\ 0, & (x_i, x_j) \in T_C \end{cases} \]

**Theorem 1.** Assume that \( IS \) is an incomplete decision system. By combining with Definition 11 and 12, we can get the incremental relational matrix:

\[ (P_{U \cup U_x}^{T})_{\text{rer}(\text{rer})} = \begin{bmatrix} (M_{U}^{T})_{\text{rer}} & (P_{U \cup U_x}^{T})_{\text{rer}(\text{rer})} \\ (P_{U \cup U_x}^{T})_{\text{rer}} & (S_{U_x}^{T})_{\text{rer}} \end{bmatrix} \]

**Theorem 2.** Assume that \( IS \) is an incomplete decision system. According to Definition 6, \( \text{GK}_{U \cup U_x}(D \mid C) \) can be defined as below:
\[ \text{Def} 6: \quad \frac{1}{|U \cup U_X|} (|U|^2 \text{GK}_{D,C}(D|C) + |U_X|^2 \text{GK}_{E_x}(D|C) + 2\sum(p_{r,t}^{GK,D})_{t,n} - 2\sum(p_{r,t}^{GK,E})_{t,n}) \]

\[ \text{Theorem 3}: \quad \text{Assume that IS is an incomplete decision system. \( \forall a \in C \), combined with Definition 6, then the new internal attribute importance is defined as:} \]
\[ \text{Si}_{\text{GK}_{U}}^{\text{inner}} (a, C, D) = \frac{1}{|U \cup U_X|} (|U|^2 \text{Si}_{\text{GK}_{U}}^{\text{inner}} (a, C, D) + |U_X|^2 \text{Si}_{\text{GK}_{U}}^{\text{inner}} (a, C, D) + 2\sum(p_{r,t}^{\text{GK,E}})_{t,n} - 2\sum(p_{r,t}^{\text{GK,D}})_{t,n} \]}

\section{3.2. The Algorithm of Adding Object Sets}

\textbf{Algorithm 2}. Incremental attribute reduction method is on account of knowledge granularity matrix in incomplete decision system (KGMIRA)

\begin{itemize}
  \item Input: \( IS \), attribute reduction when no new object set added \( RED, U_X \).
  \item Output: attribute reduction after adding objects \( RED \).
  \item (a) \( RED \rightarrow W \), calculate each incremental relational matrix.
  \item (b) Calculate \( \text{GK}_{U \cup X}(D|W) \) and \( \text{GK}_{U \cup X}(D|C) \).
  \item (c) While \( \text{GK}_{U \cup X}(D|W) = \text{GK}_{U \cup X}(D|C) \), jump to (e), otherwise execute (d).
  \item (d) When \( \text{GK}_{U \cup X}(D|W) \neq \text{GK}_{U \cup X}(D|C) \), \( \forall a \in (C - W) \), calculate \( \text{Si}_{\text{GK}_{U \cup X}}^{\text{inner}} (a, W \cup \{a\}, D) \), then add the highest significance of attribute to the \( W \), until \( \text{GK}_{U \cup X}(D|W) = \text{GK}_{U \cup X}(D|C) \).
  \item (e) \( \forall a \in W \), if \( \text{GK}_{U \cup X}(D|(W - \{a\})) = \text{GK}_{U \cup X}(D|C) \), \( W \leftarrow W - \{a\} \), if not, retained \( W \).
  \item (f) \( RED \leftarrow W \), end.
\end{itemize}

\section{4. Experimental Analysis}

\subsection{4.1. The Introduction of Dataset and Experimental Environment}

All data was downloaded from UCI website, as table 1 shown, there are two data types. We will randomly delete 5% attribute values in complete data. And you can know the object size, the characters and the classification category in the table. The software environment is Matlab2019a. Hardware environment: processor: Intel core i7-6700, memory: 8GB. In the course of the experiment, firstly, we divided the data into ten parts with the same size, which 30% will be the basic object sets, be left 70% used as new object sets, and the time consumed by attribute reduction is recorded once every 10%. The experiment was finished when all the objects were added, then recorded the experimental results.

\subsection{4.2. Experimental Result}

For figure 1, the x-axis Indicates that each increase of 10% of data, the y-axis shows the running time after increasing the object set, the circle mark line represents the algorithm HARIDS, and the diamond mark line represents the incremental algorithm KGMIRA. It can be seen that for the two algorithms, the operation time will be longer when the number of objects increases. It can be clearly found that the calculation time of the KGMIRA is shorter than the HARIDS with the same number of objects. Because the algorithm KGMIRA doesn’t need to calculate from scratch every time when it joins the object set, which shows that the KGMIR is more efficient than the HARIDS.

The reduction number is also one of the major indexes to evaluate attribute reduction algorithm. In table 2, the attribute reduction of each data set is obtained from all objects. By comparison, it is found that the attribute reduction of most data sets obtained by two algorithms are the same. On the premise that the classification accuracy has little influence, the fewer number of attribute reduction and the higher of reduction rate, the better performance of the algorithm.
Table 1. Description of dataset.

| Datasets   | types     | objects | characters | categories |
|------------|-----------|---------|------------|------------|
| 1 Zoo      | Complete  | 101     | 16         | 7          |
| 2 Soybean  | Incomplete| 307     | 35         | 19         |
| 3 Breast Cancer | Incomplete | 699     | 9          | 2          |
| 4 Mushroom | Incomplete| 8124    | 22         | 2          |
| 5 Letter   | Complete  | 20000   | 16         | 26         |

Table 2. Comparison of two algorithm on reduction.

| Data sets | HARIDS | KGMIRA |
|-----------|--------|--------|
|           | Reduction | Time (s) | Reduction | Time (s) |
| 1 Zoo     | 6,13,3,8,4 | 0.042 | 13,3,6,8,4 | 0.0086 |
| 2 Soybean | 7,16,1,22,6,15,8,4,9 | 1.531 | 16,1,7,22,6,15,8,4,9 | 0.166 |
| 3 Breast Cancer | 6,8,7,1,4,2,5 | 0.684 | 6,8,3,5,2,4,7,1 | 0.103 |
| 4 Mushroom | 2,3,4,6,10,13,14,15,16,21,22,5 | 218.6 | 2,3,4,6,10,13,14,15,16,21,22,5 | 46.36 |
| 5 Letter   | 4,8,15,9,11,13,12,10,16,5,2,1 | 1538.83 | 8,5,2,9,11,12,13,10,16,15,1,4 | 253.96 |

Figure 1. Comparison of two algorithm on computational time.

In table 3, the Naive Bayesian Classification Model algorithm in Weka software is used to calculate the classification accuracy, and the cross-validation method is adopted. The classification accuracy in the table is expressed as a percentage. Clearly, the average classification accuracy of the two algorithms under five data sets is very close or even the same, because the attribute reduction obtained by them is roughly the same. For third data, the KGMIRA’s accuracy is improved compared with the HARIDS. From perspective of classification, the KGMIRA can obtain better attribute reduction in shorter time.
Table 3. Classification accuracy for HARIDS and KGMIRA.

| Datasets        | Naive Bayes Classifier |
|-----------------|------------------------|
|                 | HARIDS     | KGMIRA      |
| 1 Zoo           | 96.32      | 96.32       |
| 2 Soybean       | 63.16      | 63.16       |
| 3 Breast Cancer | 96.00      | 96.35       |
| 4 Mushroom      | 81.93      | 81.93       |
| 5 Letter        | 57.79      | 57.79       |

5. Concluding Remarks
In fact, the data in the database may increase dynamically by row. The algorithm put forward in this text is applicable and can effectively deal with incomplete data when the object set changes dynamically. However, considering that the matrix needs to occupy a lot of storage space, which takes much time to calculate. So, we will focus on the practical application to verify the proposed methods and study how to improve their efficiency through parallel strategy in future. At the same time, we intend to study the incremental algorithm in the form of non-matrix under knowledge granularity, then the proposed algorithm is extended to other rough set models to further study the attribute reduction method when the attribute sets and attribute values change dynamically.

Acknowledgements
This research is supported by Science and Technology Project of Jiangxi Provincial Department of Education (GJJ170995) and National Natural Science Foundation of China (61562061).

References
[1] Pawlak Z 1991 (Dordrecht: Kluwer Academic Publishers).
[2] Wang X, Tsang E C C, Zhao S and Chen D 2007 Inf. Sci. 177 (20) 4493-4514.
[3] Ni P, Zhao S, Wang X, Chen H and Li C PARA 2019 Inf. Sci. 503 533-550.
[4] Wang F, Liang J and Dang C 2013 Appl Soft Computer 13 (1) 676-689.
[5] Konecny J and Krajca 2018 Inf. Sci. 467 431-445.
[6] Yang Y, Chen D and Wang H 2017 IEEE Trans Fuzzy Syst 25 (4) 825-838.
[7] Jing Y G, Li T R, Luo C, et al. 2016 Knowledge-Based Systems 104 24-38.
[8] Shu W and Shen H 2014 Pattern Recognition 47 (12) 3890-3906.
[9] Luo C, Li T and Yao Y 2017 Inf. Sci. 417 39-54.
[10] Ding M W and Zhang T F 2017 Computer Science 44 (7) 244-250.