Intelligent Automation & Soft Computing
Tech Science Press

DOI: 10.32604/iasc.2023.039553
Article

An Update Method of Decision Implication Canonical Basis on Attribute Granulating

Yanhui Zhai[1,2,*], Rujie Chen[1] and Deyu Li[1,2]

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

*Corresponding Author: Yanhui Zhai. Email: chai_yanhui@163.com
Received: 04 February 2023; Accepted: 11 April 2023; Published: 23 June 2023

Abstract: Decision implication is a form of decision knowledge representation, which is able to avoid generating attribute implications that occur between condition attributes and between decision attributes. Compared with other forms of decision knowledge representation, decision implication has a stronger knowledge representation capability. Attribute granularization may facilitate the knowledge extraction of different attribute granularity layers and thus is of application significance. Decision implication canonical basis (DICB) is the most compact set of decision implications, which can efficiently represent all knowledge in the decision context. In order to mine all decision information on decision context under attribute granulating, this paper proposes an updated method of DICB. To this end, the paper reduces the update of DICB to the updates of decision premises after deleting an attribute and after adding granulation attributes of some attributes. Based on this, the paper analyzes the changes of decision premises, examines the properties of decision premises, designs an algorithm for incrementally generating DICB, and verifies its effectiveness through experiments. In real life, by using the updated algorithm of DICB, users may obtain all decision knowledge on decision context after attribute granularization.

Keywords: Decision context; attribute granulating; decision implication; decision implication canonical basis

1 Introduction

Formal Concept Analysis (FCA) is a data analysis and processing tool proposed by Prof. Wille [1]. FCA has been widely studied [2–5] and applied in machine learning [6–8], data mining [9,10], information retrieval [11,12], conflict analysis [13–15] and recommendation systems [16–18].

In FCA, a formal context is a two-dimensional table that reflects relationships between objects and attributes. Attribute implication is a formal representation of knowledge in formal contexts. Since the number of (attribute) implications extracted from formal context is very large, Qu et al. [19] proposed decision implication to reduce the implications that occur between condition attributes or between decision attributes. Zhai et al. [20] studied the semantical and syntactical characteristics of decision implication from a logical perspective. In particular, in the syntactical aspect, Zhai et al. [20] proposed two inference rules, namely AUGMENTATION and COMBINATION, to perform knowledge reasoning among decision implications. Zhai et al. [21] further investigated the influences of the order and number of times of applying inference rules on knowledge reasoning and designed an optimal reasoning strategy.

The number of decision implications in decision contexts, however, is still huge. Thus, Zhai et al. [22] introduced decision premise and decision implication canonical basis (DICB), and proved that DICB is a complete, non-redundant and optimal set of decision implications. In other words, DICB can not only keep all the information in decision contexts (i.e., completeness), but also contain the least number of decision implications among all complete sets of decision implications (optimality). To generate DICB efficiently, Li et al. [23] proposed a true premise-based generation algorithm for DICB, and Zhang et al. [24] proposed an incremental generation algorithm for DICB.

In fact, researchers have investigated other forms of knowledge representation and reasoning besides decision implication, such as concept rules and granular rules [25,26]. Concept rules are decision implications where both premises and consequences are intents [25], and granular rules are decision implications where both premises and consequences are granular intents [25]. Based on these works, Qin et al. [27] investigated attribute (object)-oriented decision rule acquisition; Xie et al. [28] discussed decision rule acquisition in multi-granularity decision context; Zhi et al. [29] proposed a fuzzy rule acquisition method based on granule description in fuzzy formal context; Hu et al. [30] studied rules whose premises and consequences are dual (formal) concepts and formal (dual) concepts. Zhang et al. [31] compared concept rule, granular rule and decision implication, and found that concept rule has stronger knowledge representation capability than granular rule and decision implication has stronger knowledge representation capability than granular rule and concept rule, i.e., decision implication is the strongest form of knowledge representation and reasoning on decision context. Thus, it is recommended in [21] that a knowledge representation and reasoning system based on decision implication can be constructed in applications by using DICB as the knowledge base and CON-COMBINATION and AUGMENTATION as the inference engine [21].

On the other hand, since granular computing [32] is able to solve complex problems with multi-level structures and facilitate knowledge acquisition at different levels and granulations, many granulation methods have been developed in FCA such as relational granulation [33,34], attribute granulation [35–41], and object granulation [42]. This paper mainly focuses on knowledge discovery on attribute granulating.

The existing attribute granulation methods mainly focus on how to update concept lattice after transforming attribute granulation [35], aiming at reducing the complexity of regenerating concept lattice. For example, Belohlavek et al., Wan et al. and Zou et al. proposed some algorithms to update concept lattice on attribute granulating [35–37]; Shao et al. [38] proposed an algorithm to update object(attribute)-oriented multi-granularity concept lattice.

The existing studies did not take into account knowledge discovery on attribute granulating. Because decision implication is superior to concept rule and granular rule, the paper will examine the update of decision implication on attribute granulating; in particular, since DICB can efficiently represent all the information of decision implication [22], it is sufficient to examine the update of DICB. Thus, the aim of the paper is to examine the relationship between DICBs before and after granulation and find a more efficient method to generate the DICB of the granulation context.

This paper is organized as follows. Section 2 reviews the related concepts and properties of decision implication and DICB; Section 3 introduces granulation context and discusses the updates of decision premises (the premises of DICB) on attribute granulating; Section 4 proposes an incremental method for updating DICB; Section 5 concludes the paper.

2 Basic Notions

This section reviews the basic concepts and conclusions of decision implication and DICB; further details can be found in [19,20].

Definition 1 [19]: Decision context K is a triple $K = (G, C \cup D, I_C \cup I_D)$, where G denotes the set of objects, C denotes the set of condition attributes, D denotes the set of decision attributes, $C \cap D = \emptyset$, $I_C \subseteq G \times C$ denotes the set of incidence relations between objects and condition attributes, and $I_D \subseteq G \times D$ denotes the set of incidence relations between objects and decision attributes. For $g \in G, m \in C$ or $m \in D$, $(g, m) \in I_C$ or $(g, m) \in I_D$ means "object g has attribute m".

Definition 2 [19]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $A \subseteq G$, $B_1 \subseteq C$ and $B_2 \subseteq D$, we define:

(1) $A^C = \{m \in C |\ (g, m) \in I_C, \forall g \in A\}$
(2) $A^D = \{m \in D |\ (g, m) \in I_D, \forall g \in A\}$
(3) ${B_1}^C = \{g \in G | (g, m) \in I_C, \forall m \in B_1\}$
(4) ${B_2}^D = \{g \in G | (g, m) \in I_D, \forall m \in B_2\}$

Proposition 1 [19]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $A_1, A_2 \subseteq C$ and $B_1, B_2 \subseteq D$, we have:

(1) $A_1 \subseteq A_2 \Longrightarrow A_2^C \subseteq A_1^C$, $B_1 \subseteq B_2 \Longrightarrow B_2^D \subseteq B_1^D$
(2) $(A_1 \cup A_2)^C = A_1^C \cap A_2^C$, $(B_1 \cup B_2)^D = B_1^D \cap B_2^D$
(3) $(A_1 \cap A_2)^C \supseteq A_1^C \cup A_2^C$, $(B_1 \cap B_2)^D \supseteq B_1^D \cup B_2^D$

Definition 3 [20]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $A \subseteq C$ and $B \subseteq D$, if $A^C \subseteq B^D$, then $A \Rightarrow B$ is called a decision implication of K, where A is the premise of $A \Rightarrow B$ and B is the conclusion of $A \Rightarrow B$.

Definition 4 [22]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. A set $A \subseteq C$ is called a decision premise of K, if A satisfies the following conditions:

(1) A is minimal with respect to A^{CD}, i.e., if $A_i \subset A$, then ${A_i}^{CD} \subsetneq A^{CD}$;
(2) A is proper, i.e.,

$$A^{CD} \supset \cup \{A_i^{CD} | A_i \subset A \text{ is a decision premise of } K\} = \cup \{A_i^{CD} | A_i \subset A\} \tag{1}$$

The set

$$K_{\mathbb{D}}^* = \cup \{A \Rightarrow A^{CD} | A \text{ is a decision premise of } K\} \tag{2}$$

is called the decision implication canonical basis (DICB) of K.

Zhai et al. [22] proved that for any decision context, DICB is complete, non-redundant, and optimal. In other words, DICB can be considered as a complete (completeness) and compact (non-redundancy and optimality) knowledge representation in decision context. Furthermore, Li et al. [23] proved that the properness of decision premise implies minimality, i.e., A is a decision premise if and only if A is proper.

3 DICB on Attribute Granulating

This section introduces granulation context based on attribute granularity refinement [39] and studies the update of decision premises.

Definition 5: Let $K = (G, C \cup D, I_C \cup I_D)$ be a formal context and $a \in C$. The attribute a can be refined to a set of attributes $\{a_1, a_2, \ldots, a_n\}$, where $a_i^C \neq \varnothing$, $a_i^C \cap a_j^C = \varnothing$ for $i \neq j$, and $\cup_{i=1}^n a_i^C = a^C$. The decision context $K_a = \left(G, C_a \cup D, I_C^a \cup I_D\right)$ is called a granulation context of K, where $C_a = C \backslash \{a\} \cup \{a_1, a_2, \ldots, a_n\}$ and $I_C^a = \{(g, m) | g \in G, m \in C_a, g \in m^C\}$.

It can be seen from Definition 5 that the refinement of a divides a into n mutually exclusive attributes $\{a_1, a_2, \ldots, a_n\}$, and any object cannot have any two of them at the same time. In other words, a_i^C, $i = 1, 2, \ldots, n$ constitute a partition of a^C. Obviously, we have $1 < n \leq |a^C|$. In granulation context, since the attribute a has been refined, C_a thus removes a from C and adds the refined attributes $\{a_1, a_2, \ldots, a_n\}$.

Example 1. A decision context for bank marketing is shown in Table 1, where $G = \{g_1, g_2, g_3, g_4, g_5, g_6, g_7, g_8, g_9\}$ containing nine customers, and $C = \{a, b, c, d, e, f, g\}$ and $D = \{h, i\}$ representing the features of the customers, where a stands for whether the customer has a job, b stands for whether the customer is married, c stands for whether the customer has a bachelor's degree or above, d stands for whether the customer has a credit default, e stands for whether the customer has a home loan, f stands for whether the customer has a personal loan, g stands for whether the customer has contacted other customers within six months, h stands for whether the customer has applied for a loan, and i stands for whether the customer will make a fixed deposit.

Table 1: Decision context for bank marketing

	a	b	c	d	e	f	g	h	i
g_1	1	1	1	0	0	0	1	0	1
g_2	1	1	0	0	1	0	1	1	0
g_3	1	0	1	0	1	0	0	1	0
g_4	1	1	1	0	0	1	1	1	0
g_5	1	0	1	0	0	0	0	0	1
g_6	1	0	1	0	1	0	1	1	0
g_7	1	0	1	0	1	0	0	0	0
g_8	1	1	0	0	0	0	0	1	0
g_9	0	1	0	1	1	0	0	0	0

When attribute a is granulated, "has a job" is granulated into "doctor", "teacher", "civil servant", "self-employed" and "other professions", recorded as, a_1: doctor, a_2: teacher, a_3: civil servant, a_4: self-employed, a_5: other profession.

The granulation context is shown in Table 2, where $G = \{g_1, g_2, g_3, g_4, g_5, g_6, g_7, g_8, g_9\}$, $C_a = C \backslash \{a\} \cup \{a_1, a_2, a_3, a_4, a_5\} = \{b, c, d, e, f, g\} \cup \{a_1, a_2, a_3, a_4, a_5\} = \{a_1, a_2, a_3, a_4, a_5, b, c, d, e, f, g\}$, and $D = \{h, i\}$.

Table 2: The granulation context

	a_1	a_2	a_3	a_4	a_5	b	c	d	e	f	g	h	i
g_1	1	0	0	0	0	1	1	0	0	0	1	0	1
g_2	0	1	0	0	0	1	0	0	1	0	1	1	0
g_3	0	0	1	0	0	0	1	0	1	0	0	1	0
g_4	0	0	1	0	0	1	1	0	0	1	1	1	0
g_5	0	0	0	0	1	0	1	0	0	0	0	0	1
g_6	0	1	0	0	0	0	1	0	1	0	1	1	0
g_7	1	0	0	0	0	0	1	0	1	0	0	0	0
g_8	0	0	0	1	0	1	0	0	0	0	0	1	0
g_9	0	0	0	0	0	1	0	1	1	0	0	0	0

According to [22], DICB can retain all the knowledge in decision context. Therefore, DICB after attribute granulation can also retain all the knowledge in granulation context. Furthermore, since decision premise determines DICB [22], the update of DICB can be reduced to the update of decision premises.

To simplify discussion, this paper divides the updates of decision premises into two steps, namely, the update of decision premises after deleting attribute a and the update of decision premises after adding granulation attributes $\{a_1, a_2, \ldots, a_n\}$.

3.1 The Updates of Decision Premises after Deleting Attribute

After attribute a is removed from $K = (G, C \cup D, I_C \cup I_D)$, the new decision context is denoted as:

$$K_{\overline{a}} = (G, C_{\overline{a}} \cup D, I_{\overline{a}} \cup I_D) \tag{3}$$

where $C_{\overline{a}} = C\backslash\{a\}$ and $I_{\overline{a}} \subseteq G \times C_{\overline{a}} \cap I_C$. In order to avoid confusion, the operation $(.)^C$ in $K_{\overline{a}}$ is denoted as $(.)^{\overline{C}}$. Since D does not change, $(.)^D$ is applicable in $K_{\overline{a}}$.

Because decision premise is defined by the operator $(.)^{CD}$, in order to examine the update of decision premise from K to $K_{\overline{a}}$, it is necessary to examine the update from A^{CD} to $A^{\overline{CD}}$, i.e., from A^{CD} to $A^{\overline{C}D}$.

First, the following properties are given.

Proposition 2: For decision contexts K and $K_{\overline{a}}$, let $A \subseteq C$. Then, the following conclusions hold:

(1) If $a \notin A$, then $A^{\overline{C}} = A^C$, namely $A^{\overline{C}D} = A^{CD}$
(2) If $a \in A$, then $A^{\overline{C}D} \cup a^{CD} \subseteq A^{CD}$
(3) $A^{\overline{C}D} \subseteq A^{CD}$

Proof: (1) If $a \notin A$, by the definitions of $(.)^C$ and $(.)^{\overline{C}}$, we can obtain $A^{\overline{C}} = A^C$ and thus $A^{\overline{C}D} = A^{CD}$.

(2) If $a \in A$, by the definitions of $(.)^C$ and $(.)^{\overline{C}}$, we have $A^C = (A\backslash\{a\} \cup a)^C = (A\backslash\{a\})^C \cap a^C = A^{\overline{C}} \cap a^C$. Thus, we obtain $A^{CD} = (A^{\overline{C}} \cap a^C)^D \supseteq A^{\overline{C}D} \cup a^{CD}$ and thus $A^{\overline{C}D} \cup a^{CD} \subseteq A^{CD}$.

(3) It follows from (1) and (2).

In order to generate the decision premises of $K_{\overline{a}}$ by the decision premises of K, we classify the changes of decision premises from K to $K_{\overline{a}}$ into three cases:

(1) If A is a decision premise of K and A is also a decision premise of $K_{\overline{a}}$, A is an unchanged decision premise.
(2) If A is a decision premise of K and A is not a decision premise of $K_{\overline{a}}$, A is an invalid decision premise.
(3) If A is not a decision premise of K and A is a decision premise of $K_{\overline{a}}$, A is a new decision premise.

The following conclusion identifies the characteristics of unchanged decision premises.

Proposition 3: For decision contexts K and $K_{\overline{a}}$, if A is a decision premise of K, then A is an unchanged decision premise if and only if $a \notin A$.

Proof: Sufficiency: By Proposition 2, if $a \notin A$, then we have $A^{\overline{C}D} = A^{CD}$. Furthermore, for A_j such that $A_j \subset A$, since $a \notin A_j$, we have $A_j^{\overline{C}D}=A_j^{CD}$. Because A is a decision premise of K, we have $A^{\overline{C}D} = A^{CD} \supset \cup\left\{A_j^{CD}|A_j \subset A\right\} = \cup\left\{A_j^{\overline{C}D}|A_j \subset A\right\}$. According to Definition 4, A is a decision premise of $K_{\overline{a}}$.

Necessity: Since A is an unchanged decision premise, A is a decision premise of $K_{\overline{a}}$.

The following conclusion identifies the characteristics of invalid decision premises.

Proposition 4: For decision contexts K and $K_{\overline{a}}$, if A is a decision premise of K, then A is an invalid decision premise if and only if $a \in A$.

Proof: If A is a decision premise of K, then A is an invalid decision premise or an unchanged decision premise. In this case, if A is an invalid decision premise, A is not an unchanged decision premise and by Proposition 3, we have $a \in A$. Conversely, if $a \in A$, by Proposition 3, A is not an unchanged decision premise and A must be an invalid decision premise.

The following result shows that there does not exist new decision premise in $K_{\overline{a}}$.

Proposition 5: For decision context $K_{\overline{a}}$, there does not exist new decision premise.

Proof: Assume that A is a new decision premise of $K_{\overline{a}}$. Since A is a decision premise of $K_{\overline{a}}$, by the definition of $K_{\overline{a}}$, we have $a \notin A$ and by Proposition 2, we have $A^{\overline{C}} = A^{C}$. Thus, there does not exist new decision premise in $K_{\overline{a}}$.

It can be seen from Proposition 5 that decision premises from K to $K_{\overline{a}}$ can be divided into two categories: unchanged decision premise and invalid decision premise. Because the invalid decision premises do not hold in $K_{\overline{a}}$, the decision premises in $K_{\overline{a}}$ only contain the unchanged decision premises of K. In other words, no new decision premises should be added after removing the condition attribute, and new decision premises appear after adding granulation attributes.

3.2 The Updates of Decision Premise after Adding Granulation Attributes

This section examines the update of decision premise in K_a from decision premise in $K_{\overline{a}}$ after adding the granulation attributes $\{a_1, a_2, \ldots, a_n\}$ to $K_{\overline{a}}$.

Similarly, we classify the changes of decision premises from $K_{\overline{a}}$ to K_a into three cases:

(1) If A is a decision premise of $K_{\overline{a}}$ and A is also a decision premise of K_a, A is an unchanged decision premise.
(2) If A is a decision premise of $K_{\overline{a}}$ and A is not a decision premise of K_a, A is an invalid decision premise.

(3) If A is not a decision premise of $K_{\overline{a}}$ and A is a decision premise of K_a, A is a new decision premise.

In order to avoid confusion, the operation $(.)^C$ in K_a is denoted as $(.)^{\tilde{C}}$. Since D does not change, $(.)^D$ is applicable in K_a. The properties of $(.)^{\overline{C}}$ and $(.)^{\tilde{C}}$ in $K_{\overline{a}}$ and K_a are given below.

Proposition 6: For decision contexts $K_{\overline{a}}$ and K_a, and $A \subseteq C_a$, the following conclusions hold:

(1) If $A \subseteq C_{\overline{a}}$, then $A^{\overline{C}} = A^{\tilde{C}}$, $A^{\overline{C}D} = A^{\tilde{C}D}$
(2) If $A \nsubseteq C_{\overline{a}}$, then $(A \cap C_{\overline{a}})^{\overline{C}D} \subseteq A^{\tilde{C}D}$.

Proof: (1) By the definitions of $(.)^{\tilde{C}}$ and $(.)^{\overline{C}}$, it is obvious.

(2) By (1), we have $(A \cap C_{\overline{a}})^{\overline{C}D} = (A \cap C_{\overline{a}})^{\tilde{C}D}$, and by $A \cap C_{\overline{a}} \subseteq A$, we have $(A \cap C_{\overline{a}})^{\tilde{C}D} \subseteq A^{\tilde{C}D}$ and thus $(A \cap C_{\overline{a}})^{\overline{C}D} \subseteq A^{\tilde{C}D}$.

The following conclusion identifies the characteristics of unchanged and invalid decision premises from $K_{\overline{a}}$ to K_a.

Proposition 7: For decision contexts $K_{\overline{a}}$ and K_a, if A is a decision premise of $K_{\overline{a}}$, then A is a decision premise of K_a.

Proof: Since A is a decision premise of $K_{\overline{a}}$, we have $A \subseteq C_{\overline{a}} \subseteq C_a$ and by Proposition 6, we have $A^{\tilde{C}D} = A^{\overline{C}D}$; furthermore, for A_j satisfying $A_j \subset A \subseteq C_{\overline{a}} \subseteq C_a$, we have $A_j^{\tilde{C}D} = A_j^{\overline{C}D}$ and thus $A^{\tilde{C}D} = A^{\overline{C}D} \supset \bigcup \left\{A_j^{\overline{C}D} | A_j \subset A\right\} = \bigcup \left\{A_j^{\tilde{C}D} | A_j \subset A\right\}$. It can be seen from Definition 4 that A is a decision premise of $K_{\overline{a}}$.

Proposition 7 shows that all the decision premises of $K_{\overline{a}}$ are the unchanged decision premises of K_a. Thus, according to the classification of decision premises of K_a, there does not exist invalid decision premise from $K_{\overline{a}}$ to K_a.

Next, we discuss the new decision premises in K_a.

Proposition 8: If A is a new decision premise from $K_{\overline{a}}$ to K_a, then there exists only one $a_i \in \{a_1, a_2, \ldots, a_n\}$ such that $a_i \in A$.

Proof: First, we will prove that there exists $a_i \in \{a_1, a_2, \ldots, a_n\}$ such that $a_i \in A$. Since A is a new decision premise from $K_{\overline{a}}$ to K_a, A is not a decision premise of $K_{\overline{a}}$ and A is a decision premise of K_a, i.e., we have $A^{\overline{C}D} = \bigcup \left\{A_j^{\overline{C}D} | A_j \subset A\right\}$ and $A^{\tilde{C}D} \supset \bigcup \left\{A_j^{\tilde{C}D} | A_j \subset A\right\}$. If there does not exist $a_i \in \{a_1, a_2, \ldots, a_n\}$ such that $a_i \in A$, we have $A \subseteq C_{\overline{a}}$. By Proposition 6, we have $A^{\tilde{C}D} = A^{\overline{C}D}$; similarly, for A_j satisfying $A_j \subset A \subseteq C_{\overline{a}} \subseteq C_a$, we have $A_j^{\tilde{C}D} = A_j^{\overline{C}D}$ and thus $A^{\tilde{C}D} = A^{\overline{C}D} = \bigcup \left\{A_j^{\overline{C}D} | A_j \subset A\right\} = \bigcup \left\{A_j^{\tilde{C}D} | A_j \subset A\right\}$, which contradicts with $A^{\tilde{C}D} \supset \bigcup \left\{A_j^{\tilde{C}D} | A_j \subset A\right\}$. Thus, we have $A \nsubseteq C_{\overline{a}}$, and there must exist $a_i \in \{a_1, a_2, \ldots, a_n\}$ such that $a_i \in A$.

Next, suppose that more than one attribute in $\{a_1, a_2, \ldots, a_n\}$ may be contained in A, say, $\left\{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\right\} \subseteq A$, $2 \leq k \leq n$.

By the definition of redundant decision implication, a decision implication $E \Rightarrow F$ is redundant with respect to a set L of decision implications if and only if for any $T \subseteq C_a \cup D$, if T satisfies L, then T also satisfies $E \Rightarrow F$. Therefore, if any $T \subseteq C_a \cup D$ satisfies $E \Rightarrow F$, then $E \Rightarrow F$ is redundant with respect to any L.

For any $T \subseteq C_a \cup D$, by the definition of granulation attributes, there exists at most one $a_k \in \{a_1, a_2, \ldots, a_n\}$ such that $a_k \in T$. Therefore, we have $\left\{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\right\} \nsubseteq T$, and since $\left\{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\right\} \subseteq A$, we have $A \nsubseteq T$, i.e., T satisfies $A \Rightarrow A^{\tilde{C}D}$. Thus, $A \Rightarrow A^{\tilde{C}D}$ is a redundant decision implication of

K_a. Since DICB is non-redundant, A is not a decision premise, which contradicts with the fact that A is a new decision premise of K_a.

By the discussion in Sections 3.1 and 3.2, for a decision premise A of K, if $a \notin A$, by Proposition 3 and Proposition 7, A must be a decision premise of $K_{\overline{a}}$ and also a decision premise of K_a. In addition, by Proposition 7 and 8, the decision premises of K_a also include the new decision premises of the form $\{a_i\} \cup E$, where $a_i \in \{a_1, a_2, \ldots, a_n\}$ and $E \subseteq C_{\overline{a}}$, which can be generated based on true premise, as shown in Section 4.

4 Algorithm for Updating DICB on Attribute Granulating

By Section 3, some decision premises of K_a can be directly obtained by judging whether the attribute a is contained in the decision premises A of K, and other decision premises of K_a is of the form $\{a_i\} \cup E$. By Definition 4, in order to determine whether $\{a_i\} \cup E$ is a decision premise, it is necessary to determine whether each subset of $\{a_i\} \cup E$ is a decision premise. Due to the complexity of enumerating all the subsets of $\{a_i\} \cup E$, we propose a true premise [23] based algorithm for determining whether $\{a_i\} \cup E$ is a decision premise, as well as updating DICB.

Firstly, Definition 6 gives the definition of true premise.

Definition 6 [23]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $A \subseteq C$ and $B \subseteq D$, A is a true premises of B, if A satisfies the following conditions:

(1) A is a premise of B (i.e., $A^C \subseteq B^D$).
(2) For any $A_i \subset A$, A_i is not a true premise of B.

The following theorem shows the equivalence of the decision premise and the true premise.

Theorem 1 [23]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context and $A \subseteq C$. Then, A is a decision premise if and only if A is a true premise of some $d \in A^{CD}$.

According to Theorem 1, we can determine whether $\{a_i\} \cup E$ is a decision premise of K_a by determining whether $\{a_i\} \cup E$ is a true premise of some decision attributes. In other words, in order to generate the decision premises of the form $\{a_i\} \cup E$, it is sufficient to generate all the true premise of some $d \in D$ that contains a_i.

Definition 7 defines the symbols needed to calculate true premise.

Definition 7 [23]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $d \in D$ and $g \in G$, denote $d^{\notin} = \{g \in G | d \notin g^D\}$ and define $g \swarrow d$ if the following conditions are satisfied:

(1) $g \in d^{\notin}$;
(2) For $h \in G$, if $g^C \subset h^C$, then $d \in h^D$.

For $d \in D$, we denote $\Phi(d) = \{g \in G | g \swarrow d\}$.

Definition 8 gives the concepts of candidate premise and candidate true premise, which are used in the process of generating true premise.

Definition 8 [23]: Let $K = (G, C \cup D, I_C \cup I_D)$ is a decision context. For $A \subseteq C$, $d \in D$ and $P \subseteq \Phi(d)$, we define:

(1) If $P = \varnothing$, or for any $g \in P$, we have $A \nsubseteq g^C$, then A is called a candidate premise of d under P, denoted by $A \to_P d$; otherwise, we denote $A \nrightarrow_P d$.
(2) If $A \to_P d$, and for any $A_i \subset A$, we have $A_i \nrightarrow_P d$, then A is called a candidate true premise of d under P, denoted by $A \cdot \to_P d$; otherwise, we denote $A \cdot \nrightarrow_P d$.

Proposition 9 [23]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $A \subseteq C$ and $d \in D$, we have:

(1) If $A \rightarrow_{\Phi(d)} d$, then A is a premise of d.
(2) If $A \cdot\!\rightarrow_{\Phi(d)} d$, then A is a true premise of d.

By Definition 7 and Proposition 9, true premise can be identified based on $\Phi(d)$ and on $d^{\notin}$. Since $d^{\notin}$ is unchanged on attribute granulating, we need to discuss the change of $\Phi(d)$ on attribute granulating. In K_a, $\Phi(d)$ is denoted as $\Phi'(d)$.

Proposition 10: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context and $K_a = \left(G, C_a \cup D, I_C^a \cup I_D\right)$ be a granulation context of K, and $d \in D$. Then we have:

(1) $\Phi(d) \subseteq \Phi'(d)$.
(2) $\Phi'(d) = \Phi(d) \cup \{g \in (a^C \cap d^{\notin}) \backslash \Phi(d) \,|\text{For any } h \in d^{\notin} \text{such that } h^C \supset g^C, \text{there does not exist } a_i \in C_a \text{such that } \{g, h\} \subseteq a_i^{\tilde{C}}\}$.

Proof: (1) By the definitions of $\Phi(d)$ and $\Phi'(d)$, it is sufficient to prove that for $g \in \Phi(d)$, there does not exist $h \in d^{\notin}$ in K_a such that $g^{\tilde{C}} \subset h^{\tilde{C}}$.

If $g \notin a^C$, by the granulation process of a in Definition 5, for any $a_i \in C_a$, we have $g \notin a_i^{\tilde{C}}$, i.e., $g^C = g^{\tilde{C}}$. In K_a, for $h \in d^{\notin}$, we need to consider two cases: (a) If for any $a_i \in C_a$, we have $h \notin a_i^{\tilde{C}}$, we obtain $h^C = h^{\tilde{C}}$. From $g \in \Phi(d)$, it follows that there does not exist $h \in d^{\notin}$ such that $g^C \subset h^C$ in K, i.e., there does not exist $h \in d^{\notin}$ such that $g^{\tilde{C}} \subset h^{\tilde{C}}$. (b) If there exists $a_i \in C_a$ such that $h \in a_i^{\tilde{C}}$, by Definition 5, there exists only one $a_i \in C_a$ such that $h \in a_i^{\tilde{C}}$. Then, we have $h^{\tilde{C}} = h^C \backslash \{a\} \cup \{a_i\}$. Assume $g^C = g^{\tilde{C}} \subset h^{\tilde{C}}$. By $g \notin a^C$, it is easy to prove that $g^C \subseteq h^C \backslash \{a\} \subset h^C$. By the definition of $\Phi(d)$ and $g \in \Phi(d)$, we have $d \in h^D$ and thus $h \notin d^{\notin}$, which contradicts with $h \in d^{\notin}$. Therefore, there does not exist $h \in d^{\notin}$ such that $g^{\tilde{C}} \subset h^{\tilde{C}}$.

If $g \in a^C$, we assume $g \notin \Phi'(d)$, i.e., there exists $h \in d^{\notin}$ in K_a such that $g^{\tilde{C}} \subset h^{\tilde{C}}$. By $g \in a^C$ and the granulation process of a, there exists only one $a_i \in C_a$ such that $g \in a_i^{\tilde{C}}$, and since $g^{\tilde{C}} \subset h^{\tilde{C}}$, we have $h \in a_i^{\tilde{C}}$ and thus $\{g, h\} \subseteq a_i^{\tilde{C}}$. Since $a_i^{\tilde{C}} \subset a^C$, we have $\{g, h\} \subset a^C$. From $h^{\tilde{C}} = h^C \backslash \{a\} \cup \{a_i\}$ and $g^{\tilde{C}} = g^C \backslash \{a\} \cup \{a_i\}$, it follows $h^C \backslash \{a\} \cup \{a_i\} \supset g^C \backslash \{a\} \cup \{a_i\}$, i.e., $h^C \supset g^C$. By the definition of $\Phi(d)$ and $g \in \Phi(d)$, we have $d \in h^D$ and thus $h \notin d^{\notin}$, which contradicts with $h \in d^{\notin}$. Thus, there does not exist $h \in d^{\notin}$ such that $g^{\tilde{C}} \subset h^{\tilde{C}}$.

(2) By the definition of $\Phi'(d)$, we have $\Phi'(d) = \{g \in d^{\notin} | \text{for any } h \in d^{\notin}, \text{there is } h^{\tilde{C}} \not\supset g^{\tilde{C}}\}$. By Conclusion (1), we have $\Phi(d) \subseteq \Phi'(d) \subseteq d^{\notin}$ and thus $\Phi'(d) = \Phi(d) \cup \{g \in d^{\notin} \backslash \Phi(d) \,|\text{for any } h \in d^{\notin}$, $h^{\tilde{C}} \supset g^{\tilde{C}}$ holds$\}$.

Next, we prove that for any $g \in d^{\notin} \backslash \Phi(d)$, we have $a \in g^C$. Assume $a \notin g^C$, i.e., $g \notin a^C$. By $g \notin \Phi(d)$, there exists $h \in d^{\notin}$ in K such that $h^C \supset g^C$. We need to consider two cases. (a) If $h \in a^C$, by the granulation process of a in Definition 5, there exists only one $a_i \in C_a$ such that $h \in a_i^{\tilde{C}}$. In this case, we have $h^{\tilde{C}} = h^C \backslash \{a\} \cup \{a_i\}$. By $a \notin g^C$, we have $h^C \backslash \{a\} \supseteq g^C$ and thus $h^{\tilde{C}} \supset g^C$. By $g \notin a^C$ and the proving process of (1), it is easy to prove $g^C = g^{\tilde{C}}$, and we have $h^{\tilde{C}} \supset g^{\tilde{C}}$, i.e., h satisfies $h^{\tilde{C}} \supset g^{\tilde{C}}$ and $h \in d^{\notin}$. Thus, we have $g \notin \Phi'(d)$, which contradicts with $g \in \Phi'(d)$. Thus, we obtain $g \in a^C$. (b) If $h \notin a^C$, by $g \notin a^C$ and the proving process of (1), we have $g^C = g^{\tilde{C}}$ and $h^C = h^{\tilde{C}}$, and thus $h^{\tilde{C}} = h^C \supset g^C = g^{\tilde{C}}$. Since $h^{\tilde{C}} \supset g^{\tilde{C}}$ and $h \in d^{\notin}$, we have $g \notin \Phi'(d)$, which contradicts with $g \in \Phi'(d)$. Thus, we obtain $g \in a^C$.

Combining with $\Phi'(d) = \Phi(d) \cup \{g \in d^{\notin} \backslash \Phi(d) \,|\text{for any } h \in d^{\notin}, h^{\tilde{C}} \not\supset g^{\tilde{C}} holds\}$, we obtain $\Phi'(d) = \Phi(d) \cup \{g \in (a^C \cap d^{\notin}) \backslash \Phi(d) \,|\text{for any } h \in d^{\notin}, h^{\tilde{C}} \not\supset g^{\tilde{C}} holds\}$. It is sufficient to prove that for $g \in (a^C \cap d^{\notin}) \backslash \Phi(d)$, the following two conditions are equivalent: Condition 1: For any $h \in d^{\notin}$, h satisfies

$h^{\tilde{C}} \not\supset g^{\tilde{C}}$, and Condition 2: For any $h \in d^{\notin}$ such that $h^C \supset g^C$, there does not exist $a_i \in C_a$ such that $\{g, h\} \subseteq a_i^{\tilde{C}}$.

First, we prove that if $g \in \left(a^C \cap d^{\notin}\right) \setminus \Phi(d)$ satisfies Condition 1, then g also satisfies Condition 2. Assume that there exists $h \in d^{\notin}$ such that $h^C \supset g^C$, and that there exists $a_i \in C_a$ such that $\{g, h\} \subseteq a_i^{\tilde{C}}$. By $g \in a^C$, we have $a \in g^C \subset h^C$ and thus $\{g, h\} \subseteq a^C$. Combining $\{g, h\} \subseteq a^C$, $\{g, h\} \subseteq a_i^{\tilde{C}}$ with the granulation process of a, we obtain $h^{\tilde{C}} = h^C \setminus \{a\} \cup \{a_i\}$ and $g^{\tilde{C}} = g^C \setminus \{a\} \cup \{a_i\}$; since $h^C \supset g^C$, we have $h^{\tilde{C}} \supset g^{\tilde{C}}$. Thus, there is $h \in d^{\notin}$ such that $h^{\tilde{C}} \supset g^{\tilde{C}}$, which is contradictory to Condition 1.

Next, we prove that if $g \in \left(a^C \cap d^{\notin}\right) \setminus \Phi(d)$ satisfies Condition 2, then g also satisfies Condition 1. Assume $h \in d^{\notin}$. If $h^C \supset g^C$, according to Condition 2, there does not exist $a_i \in C_a$ such that $\{g, h\} \subseteq a_i^{\tilde{C}}$. In this case, by $g \in a^C$ and $h \in a^C$, according to the granulation process of a, there must exist $a_j, a_k \in C_a$ such that $g \in a_j^{\tilde{C}}$ and $h \in a_k^{\tilde{C}}$; because there does not exist $a_i \in C_a$ such that $\{g, h\} \subseteq a_i^{\tilde{C}}$, we have $a_j \neq a_k$, $g \notin a_k^{\tilde{C}}$, and $h \notin a_j^{\tilde{C}}$, i.e., $h^{\tilde{C}} \not\supset g^{\tilde{C}}$. If $h^C \not\supset g^C$, we have $h^C \not\supset g^C$ or $h^C = g^C$. If $h^C = g^C$, we have $h^{\tilde{C}} = g^{\tilde{C}}$, i.e., $h^{\tilde{C}} \not\supset g^{\tilde{C}}$. If $h^C \not\supset g^C$, we will prove $h^{\tilde{C}} \not\supset g^{\tilde{C}}$ in two cases. (a) If $a \in h^C$, since $h^C \not\supset g^C$, there must exist $b_1 \in C$ such that $b_1 \neq a$, $b_1 \in g^C$, and $b_1 \notin h^C$. It is easy to prove $h^{\tilde{C}} \not\supset g^{\tilde{C}}$. (b) If $a \notin h^C$, by the granulation process of a, there does not exist $a_i \in \{a_1, a_2, \ldots, a_n\}$ such that $a_i \in h^{\tilde{C}}$. By $g \in a^C$ and the granulation process of a, there is $a_i \in \{a_1, a_2, \ldots, a_n\}$ such that $a_i \in g^{\tilde{C}}$. Thus, we have $h^{\tilde{C}} \not\supset g^{\tilde{C}}$.

Next, we will generate the true premises of d that contains a_i, starting from determining whether a_i is a true premise of d.

Proposition 11: Let $K_a = \left(G, C_a \cup D, I_C^a \cup I_D\right)$ be a granulation context, $d \in D$, and $a_i \in \{a_1, a_2, \ldots, a_n\}$. Setting $P = \Phi'(d) \setminus a_i^{\tilde{C}} \neq \varnothing$, then we have $\{a_i\} \cdot \rightarrow_P d$.

Proof: For any $g \in P$, we have $a_i \notin g^{\tilde{C}}$, i.e, $\{a_i\} \nsubseteq g^{\tilde{C}}$, and thus $\{a_i\} \rightarrow_P d$. Furthermore, since $\{A_i | A_i \subset \{a_i\}\} = \{\varnothing\}$ and $\varnothing \subseteq g^{\tilde{C}}$, i.e., $\varnothing \nrightarrow_P d$, by Definition 8, we have $\{a_i\} \cdot \rightarrow_P d$.

At this time, we obtain the candidate true premise $\{a_i\}$ of d under P according to Proposition 11. Next, we need to gradually increase P to $\Phi(d)$ to determine whether $\{a_i\}$ is the true premise of d. In the incremental process, reference [23] classified candidate true premises into three cases (Definition 9) and discussed their properties (Proposition 12).

Definition 9 [23]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $A \subseteq C$, $d \in D$, $P \subseteq \Phi(d)$, and $g \in \Phi(d) \setminus P$, we define:

(1) If $A \cdot \rightarrow_P d$ and $A \cdot \rightarrow_{P \cup \{g\}} d$, then A is called an unchanged candidate true premise of d under $P \cup \{g\}$.
(2) If $A \cdot \rightarrow_P d$ and $A \cdot \nrightarrow_{P \cup \{g\}} d$, then A is called an invalid candidate true premise of d under $P \cup \{g\}$.
(3) If $A \cdot \nrightarrow_P d$ and $A \cdot \rightarrow_{P \cup \{g\}} d$, then A is called a new candidate true premise of d under $P \cup \{g\}$.

Proposition 12 [23]: Let $K = (G, C \cup D, I_C \cup I_D)$ be a decision context. For $A \subseteq C$, $d \in D$, $P \subseteq \Phi(d)$ and $g \in \Phi(d) \setminus P$, we have:

(1) A is an unchanged candidate true premise of d under $P \cup \{g\}$ if and only if $A \cdot \rightarrow_P d$ and $A \nsubseteq g^C$.
(2) A is an invalid candidate true premise of d under $P \cup \{g\}$ if and only if $A \cdot \rightarrow_P d$ and $A \subseteq g^C$.
(3) A is a new candidate true premise of d under $P \cup \{g\}$ if and only if the following conditions hold:
a) $A \cdot \nrightarrow_P d$
b) There exists an invalid candidate true premise A_m satisfying $A = A_m \cup \{a\}$ for $a \in C - g^C$
c) For any $A_i \subset A$, we have $A_i \cdot \nrightarrow_{P \cup \{g\}} d$.

By Definition 9 and Proposition 12, the objects in the set $\Phi'(d) \cap a_i^{\tilde{C}}$ can be gradually added into P. If $\{a_i\} \cdot \rightarrow_{\Phi'(d)} d$, then $\{a_i\}$ is a true premise of d; otherwise $\{a_i\}$ is not a true premise of d. After adding g in $\Phi'(d) \cap a_i^{\tilde{C}}$ to P, by Definition 9, $\{a_i\}$ may be an unchanged or an invalid candidate true premise of d under g. If $\{a_i\}$ is an unchanged candidate truth premise, we can continue to add the remaining objects in $\Phi'(d) \cap a_i^{\tilde{C}}$ to P. If $\{a_i\}$ is an invalid candidate truth premise, by Proposition 12(3), $A = \{a_i\} \cup \{b\}$ can be generated, where $b \in C_a - g^C$. In this case, it needs to further judge whether $A = \{a_i, b\}$ is a new candidate true premise of d under $P \cup \{g\}$ by Proposition 12(3). If $\{a_i, b\}$ is a new candidate true premise, the remaining objects in $\Phi'(d) \cap a_i^{\tilde{C}}$ should be gradually added to $P \cup \{g\}$ to determine whether $\{a_i, b\} \cdot \rightarrow_{\Phi'(d)} d$ holds according to the above process. By repeating the above process, one can generate all the true premises of d of the form $\{a_i\} \cup E$.

The above process can be further optimized. For example, Proposition 13 shows that if a_i satisfies some conditions, a_i is a true premise of d. In this case, except a_i, all the sets of the form $\{a_i\} \cup E\ (E \neq \emptyset)$ may not be the true premises of d, and there is no need to determine other sets $\{a_i\} \cup E$.

Proposition 13: Let $K_a = \left(G, C_a \cup D, I_C^a \cup I_D\right)$ be a granulation context, $d \in D$, $\Phi'(d) \neq \emptyset$ and $a_i \in \{a_1, a_2, \ldots, a_n\}$. If $a_i^{\tilde{C}} \cap \Phi'(d) = \emptyset$, then $\{a_i\}$ is a true premise of d.

Proof: Let $P = \left\{g | g \in \Phi'(d) \setminus a_i^{\tilde{C}}\right\}$. If $a_i^{\tilde{C}} \cap \Phi'(d) = \emptyset$, then we have $P = \Phi'(d)$. By Proposition 12, we obtain $\{a_i\} \cdot \rightarrow_P d$, i.e., $\{a_i\} \cdot \rightarrow_{\Phi'(d)} d$, and by Proposition 9, we know that a_i is a true premise of d.

A true premise-based incremental method can then be proposed for generating DICB under attribute granulation, as shown in Algorithm 1.

Algorithm 1: True premise-based algorithm for generating DICB on attribute granulating

Input: Decision context $\boldsymbol{K} = (\boldsymbol{G}, \boldsymbol{C} \cup \boldsymbol{D}, \boldsymbol{I_C} \cup \boldsymbol{I_D})$, granulation context $\boldsymbol{K_a} = \left(\boldsymbol{G}, \boldsymbol{C_a} \cup \boldsymbol{D}, \boldsymbol{I_C^a} \cup \boldsymbol{I_D}\right)$, and the set $\boldsymbol{DP_K}$ of decision premises of $\boldsymbol{K}$
Output: The DICB $\boldsymbol{O_{K_a}}$ of $\boldsymbol{K_a}$

```
O_{K_a} = ∅
for all A ∈ DP_K do:
   if a ∉ A then:
      add A ⇒ A^{C̃D} to O_{K_a}
   end if
end for
for all d ∈ D do:
   Φ′(d) = get_newgd(K, K_a, d)
   for a_i in {a_1, a_2, ..., a_n}:
      dp_{a_i} = {{a_i}} //dp_{a_i} records the true premises of d in the form {a_i} ∪ E
      if a_i^{C̃} ∩ Φ′(d) ≠ ∅ then:
         dp_{a_i} = generator_newdp(a_i, Φ′(d))
      end if
      for all A ∈ dp_{a_i} do:
         if A ⇒ A^{C̃D} not in O_{K_a} then:
            add A ⇒ A^{C̃D} to O_{K_a}
         end if
      end for
```

(Continued)

Algorithm 1: Continued

19: **end for**
20: **end for**
21: **return** O_{K_a}

In Algorithm 1, we first generate the unchanged decision premises by the decision premises of K according to the results in Section 3 (Steps 2–6), and then generate the new decision premise of K_a by generating the true premises of each d that contain a_i (Steps 7–20). In Steps 7–20, for a decision attribute d, we first generate $\Phi'(d)$ by the function *get_newgd* (Step 8), then generate the new decision premises $\{a_i\} \cup E$ for each a_i by *generator_newdp* (Steps 10–13), and finally generate the decision implications of the new decision premises (Steps 14–18). In Steps 10–13, by Proposition 13, if $a_i^{\tilde{C}} \cap \Phi'(d) = \varnothing$, only a_i is a true premise of d in the sets $\{a_i\} \cup E$. Otherwise, it is necessary to gradually add the objects in $a_i^{\tilde{C}} \cap \Phi'(d)$ by the function *generator_newdp* to obtain dp_{a_i}, the set of all the true premises of d of the form $\{a_i\} \cup E$.

Algorithm 2 presents the function *get_newgd* for generating $\Phi'(d)$.

Algorithm 2: The function *get_newgd*

Input: Decision context $K = (G, C \cup D, I_C \cup I_D)$, granulation context $K_a = \left(G, C_a \cup D, I_C^a \cup I_D\right)$, and $d \in D$
Output: $\Phi'(d)$
1: $\Phi'(d) = \varnothing$
2: $\Phi(d) = getAll\_gd(K, d)$
3: **if** $\left(a^C \cap d^{\notin}\right) \backslash \Phi(d) == \varnothing$ **then**:
4: $\Phi'(d) = \Phi(d)$
5: **else:**
6: **for all** $g \in a^C \cap d^{\notin}$ && $g \notin \Phi(d)$ **do:**
7: $\Phi'(d) = \Phi(d) \cup \{g\}$
8: **for** each $h \in d^{\notin}$ && $h^C \supset g^C$:
9: **if** there exists $a_i \in C_a$ such that $\{g, h\} \subseteq a_i^{\tilde{C}}$ then:
10: $\Phi'(d) = \Phi'(d) \backslash \{g\}$
11: **break**
12: **end if**
13: **end for**
14: **end for**
15: **end if**
16: **return** $\Phi'(d)$

Algorithm 2 generates $\Phi'(d)$ according to Proposition 10. After initializing $\Phi'(d)$ (Step 1), Algorithm 2 uses the function *getAll_gd* in [23] to generate $\Phi(d)$ of K (Step 2). If $(a^C \cap d^{\notin}) \backslash \Phi(d) = \varnothing$, by Proposition 10(2), one should keep $\Phi'(d) = \Phi(d)$ (Step 4); otherwise $\Phi'(d)$ should be generated by Proposition 10 (Steps 6–14).

Algorithm 3 presents the function *generator_newdp* to generate all the true premises of d of the form $\{a_i\} \cup E$.

Algorithm 3: The function *generator_newdp*

Input: $\boldsymbol{a_i}$ and $\boldsymbol{\Phi'}(\boldsymbol{d})$
Output: $\boldsymbol{dp_{a_i}}$
1: $\boldsymbol{dp_{a_i}} = \{\{\boldsymbol{a_i}\}\}$
2: $\boldsymbol{P} = \boldsymbol{\Phi'}(\boldsymbol{d}) \setminus \boldsymbol{a_i}^{\tilde{C}}$
3: for all $\boldsymbol{g} \in \boldsymbol{a_i}^{\tilde{C}} \cap \boldsymbol{\Phi'}(\boldsymbol{d})$ **do**:
4: $\quad \boldsymbol{P} = \boldsymbol{P} \cup \{\boldsymbol{g}\}$
5: $\quad$ for all $A \in \boldsymbol{dp_{a_i}}$ **do**:
6: $\quad\quad$ if $A \subseteq \boldsymbol{g}^{\tilde{C}}$ **then:**
7: $\quad\quad\quad$ remove $\boldsymbol{A}$ from $\boldsymbol{dp_{a_i}}$
8: $\quad\quad\quad$ for each $\boldsymbol{b} \in \boldsymbol{C_a} - \boldsymbol{g}^{\tilde{C}} - \{\boldsymbol{a_1}, \boldsymbol{a_2}, \ldots, \boldsymbol{a_n}\}$:
9: $\quad\quad\quad\quad$ add $\boldsymbol{A} \cup \{\boldsymbol{b}\}$ to $\boldsymbol{dp_{a_i}}$
10: $\quad\quad\quad\quad$ for all $A_i \subseteq A \cup \{\boldsymbol{b}\}$ **do**:
11: $\quad\quad\quad\quad\quad$ if $A_i \cdot \rightarrow_P \boldsymbol{d}$ **then**:
12: $\quad\quad\quad\quad\quad\quad$ remove $\boldsymbol{A} \cup \{\boldsymbol{b}\}$ from $\boldsymbol{dp_{a_i}}$
13: $\quad\quad\quad\quad\quad\quad$ break
14: $\quad\quad\quad\quad\quad$ end if
15: $\quad\quad\quad\quad$ end for
16: $\quad\quad\quad$ end for
17: $\quad\quad$ end if
18: $\quad$ end for
19: end for
20: return $\boldsymbol{dp_{a_i}}$

Algorithm 3 first initializes dp_{a_i} to $\{\{a_i\}\}$ according to Proposition 13 (Step 1), and initializes P to $\left\{g | g \in \Phi'(d) \setminus a_i^{\tilde{C}}\right\}$ (Step 2). Then, Algorithm 3 adds the elements in $a_i^{\tilde{C}} \cap \Phi'(d)$ to P and judges whether $\{a_i\}$ is a true premise of d according to Proposition 9 (Steps 3–19). In the process (Steps 3–19), if $\{a_i\} \nsubseteq g^{\tilde{C}}$, by Proposition 12 and Definition 9, we have $\{a_i\} \cdot \rightarrow_{P \cup \{g\}} d$, and continue to add the elements in $a_i^{\tilde{C}} \cap \Phi'(d)$ and determine whether $\{a_i\}$ is a true premise of d. If $\{a_i\} \subseteq g^{\tilde{C}}$ (Step 6), by Proposition 12, $\{a_i\}$ is an invalid candidate true premise of d under $P \cup \{g\}$. By Definition 8(1) and $g \in \Phi'(d)$, we have $\{a_i\} \nrightarrow_{\Phi'(d)} d$, and by Definition 8(2), we have $\{a_i\} \cdot \nrightarrow_{\Phi'(d)} d$. Thus, $\{a_i\}$ is not a true premise of d (step 7). In this case, according to Definition 9, $\{a_i\}$ is an invalid candidate true premise, and according to Proposition 12(3), a new candidate true premise with respect to g can be generated based on $\{a_i\}$. To this end, by Proposition 12(3)(b), $b \in C_a - g^{\tilde{C}}$ can be added to $\{a_i\}$. By Proposition 8, the decision implications whose premises contain two or more granulation attributes are redundant. Thus, it is sufficient to add $C_a - g^{\tilde{C}} - \{a_1, a_2, \ldots, a_n\}$ to $\{a_i\}$ (Steps 8–9). In order to determine whether $\{a_i\} \cup \{b\}$ is a new candidate true premise of d under $P \cup \{g\}$, one should check whether Proposition 12(3)(a) and Proposition 12(3)(c) are true. For Condition (a), $\{a_i\} \cdot \rightarrow_P d$ holds by $\{a_i\} \in dp_{a_i}$ in Step 5, and $\{a_i\} \cup \{b\} \cdot \nrightarrow_P d$ holds by Definition 8 (otherwise, $\{a_i\} \cdot \nrightarrow_P d$ holds); in other words, Condition (a) holds. Thus, Algorithm 3 simply checks whether Proposition 12(3)(c) holds in Steps 10–15. It should be noted that P is equal to $(\Phi'(d) \setminus a_i^{\tilde{C}}) \cup \{g\}$ at this iteration.

For the candidate true premise $\{a_i\} \cup \{b\}$ that satisfies Condition (c), by Proposition 12(3), $\{a_i\} \cup \{b\}$ is a new candidate true premise of d under $P \cup \{g\} = (\Phi'(d) \setminus a_i^{\tilde{C}}) \cup \{g\}$, i.e., we have $\{a_i\} \cup \{b\} \cdot \rightarrow_{(\Phi'(d) \setminus a_i^{\tilde{C}}) \cup \{g\}} d$. Therefore, in the next iteration (Steps 3–19), we only need to gradually increase the objects in $a_i^{\tilde{C}} \cap \Phi'(d) \setminus \{g\}$. Similarly, in Steps 3-19, we should first determine whether $A = \{a_i, b\}$ is an invalid candidate true premise of d under $P \cup \{g\}$ according to Proposition 12 (Step 6); if $\{a_i, b\} \subseteq g^{\tilde{C}}$, $\{a_i, b\}$ is an invalid candidate true premise of d under $P \cup \{g\}$, and should be deleted by Definitions 8(1) and 8(2) (Step 7). In this case, new candidate true premises can be generated by Steps 8–16.

Example 2. (Continuing Example 1) For the decision context K in Table 1 and the granulation context K_a in Tables 2, 3 lists the DICB of K.

Table 3: DICB of K

$\{f\} \Rightarrow \{h\}$	$\{a, b, e\} \Rightarrow \{h\}$
$\{e, g\} \Rightarrow \{h\}$	$\{e, b, c\} \Rightarrow \{h, i\}$
$\{a, d\} \Rightarrow \{h, i\}$	$\{e, f\} \Rightarrow \{h, i\}$
$\{c, d\} \Rightarrow \{h, i\}$	$\{d, f\} \Rightarrow \{h, i\}$
$\{d, g\} \Rightarrow \{h.i\}$	

According to Algorithm 1 (Steps 1–6), the unchanged decision premises can be computed as $\{\{f\}, \{e, g\}, \{c, d\}, \{d, g\}, \{e, b, c\}, \{e, f\}, \{d, f\}\}$.

Next, we calculate the true premise of $\{a_i\} \cup E$ for each decision attribute. Take the decision attribute h as an example. Firstly, we need to calculate $\Phi'(h)$ of K_a according to Algorithm 2. According to K_a, we have $h^{\notin} = \{g_1, g_5, g_7, g_9\}$, $\Phi(h) = \{g_1, g_7, g_9\}$, and $a^C = \{g_1, g_2, g_3, g_4, g_5, g_6, g_7, g_8\}$. By $\left(a^C \cap h^{\notin}\right) \setminus \Phi(h) = \{g_5\}$ and Algorithm 2, there does not exist $a_i \in C_a$ satisfying $\{g_1, g_5\} \subseteq a_i^{\tilde{C}}$ or $\{g_7, g_5\} \subseteq a_i^{\tilde{C}}$, so we have $\Phi'(h) = \Phi(h) \cup \{g_5\} = \{g_1, g_5, g_7, g_9\}$.

Next, we calculate the true premises of h with the form $\{a_i\} \cup E$. According to Table 2, we have $a_1^{\tilde{C}} = \{g_1, g_7\}$, $a_2^{\tilde{C}} = \{g_2, g_6\}$, $a_3^{\tilde{C}} = \{g_3, g_4\}$, $a_4^{\tilde{C}} = \{g_8\}$ and $a_5^{\tilde{C}} = \{g_5\}$, where $a_2^{\tilde{C}} \cap \Phi'(h) = a_3^{\tilde{C}} \cap \Phi'(h) = a_4^{\tilde{C}} \cap \Phi'(h) = \varnothing$. Thus, $\{a_2\}$,$\{a_3\}$ and $\{a_4\}$ are the true premises of h. For a_5, according to Algorithm 3, we can gradually increase $a_5^{\tilde{C}} \cap \Phi'(h) = \{g_5\}$ to $\{a_5\}$. Since $\{a_5\} \subseteq g_5^{\tilde{C}}$, we add $C_a - g_5^{\tilde{C}} - \{a_1, a_2, a_3, a_4\}$ to $\{a_5\}$ to obtain the candidate true premises $\{\{a_5, b\}, \{a_5, d\}, \{a_5, e\}, \{a_5, f\}, \{a_5, g\}\}$. However, we have $\{f\} \cdot \rightarrow_{\Phi'(h)} h$; therefore, the true premises of h containing a_5 are $\{\{a_5, b\}, \{a_5, d\}, \{a_5, e\}, \{a_5, g\}\}$. Similarly, it can be obtained that the true premises of h containing a_1 are $\{\{a_1, d\}, \{a_1, b, e\}\}$.

Similarly, we can obtain the true premises of i with the form of $\{a_i\} \cup E$, i.e., $\{\{a_1, b\}, \{a_1, f\}, \{a_1, g\}, \{a_2, e\}, \{a_2, b, c\}, \{a_3, c\}, \{a_3, e, g\}, \{a_3, e, b\}, \{a_5, b\}, \{a_5, d\}, \{a_5, e\}, \{a_5, g\}, \{a_5\}\}$.

Table 4 lists the DICB of K_a.

Table 4 DICB of K_a

$\{f\} \Rightarrow \{h\}$	$\{e, b, c\} \Rightarrow \{h, i\}$
$\{e, g\} \Rightarrow \{h\}$	$\{e, f\} \Rightarrow \{h, i\}$
$\{a_5, b\} \Rightarrow \{h, i\}$	$\{d, f\} \Rightarrow \{h, i\}$
$\{a_5, d\} \Rightarrow \{h, i\}$	$\{a_1, b\} \Rightarrow \{i\}$
$\{a_5, e\} \Rightarrow \{h, i\}$	$\{a_1, f\} \Rightarrow \{h, i\}$
$\{a_5, g\} \Rightarrow \{h, i\}$	$\{a_1, g\} \Rightarrow \{i\}$
$\{a_2\} \Rightarrow \{h\}$	$\{a_1, d\} \Rightarrow \{h, i\}$
$\{a_3\} \Rightarrow \{h\}$	$\{a_2, e\} \Rightarrow \{h, i\}$
$\{a_4\} \Rightarrow \{h\}$	$\{a_2, b, c\} \Rightarrow \{h, i\}$
$\{c, d\} \Rightarrow \{h, i\}$	$\{a_3, c\} \Rightarrow \{h, i\}$

(Continued)

Table 4 Continued

$\{d,g\} \Rightarrow \{h,i\}$	$\{a_3,e,g\} \Rightarrow \{h,i\}$
$\{a_5\} \Rightarrow \{i\}$	$\{a_3,e,b\} \Rightarrow \{h,i\}$
	$\{a_1,b,e\} \Rightarrow \{h,i\}$

5 Experimental Verification

In order to verify the effectiveness of the proposed method, we conduct some experiments in real data sets. We selected four UCI data sets, performed preprocessing such as removing missing values and normalizing continuous values, and generated the binary data sets according to the threshold of 0.5. Since the true premise-based algorithm (MBTP) in [23] is the most efficient method for generating DICB, especially when $|G|/|C|$ is less than 40, we select the MBTP algorithm as the baseline algorithm for comparison. In order to produce a stable performance, the ratio of $|G|/|C|$ is set to 20 and the number of condition attributes can be derived accordingly, as shown in Table 5.

Table 5: Information of data sets

Data set	Number of objects	Number of attributes	Number of condition attributes
bank8FM	480	27	24
Supermarket	560	99	28
Credit rating	546	273	27
Hypothyroid	1024	457	51

In the experiments, MBTP is used to directly generate the DICB of granulation context and the proposed method, called the InCremental Method (ICM), is used to generate the DICB of granulation context by updating the DICB of decision context. In order to analyze the influence of the number of granulations on the experimental results, we randomly selected about one fifth of the number of condition attributes and refined them into 2, 3 and 4 fine-grained attributes respectively.

The average consuming times of generate DICB for each refinement are summarized in Table 6.

Table 6: Average consuming times of generate DICB for each refinement

Data set	Number of fine-grained attributes	The average time of MBTP(s)	The average time of ICM(s)
	2	14.43	5.63
bank8FM	3	16.30	6.51
	4	18.39	7.12
	2	961.27	501.19
Supermarket	3	1137.81	552.50
	4	1245.00	569.93
	2	58.59	32.48
Credit rating	3	62.53	34.23
	4	66.91	38.70

(Continued)

Table 6: Continued

Data set	Number of fine-grained attributes	The average time of MBTP(s)	The average time of ICM(s)
	2	1333.86	662.99
Hypothyroid	3	1515.53	812.21
	4	1698.19	962.10

According to Table 6, on the whole, ICM is more efficient than MBTP. For the data sets whose DICB can be generated within about one minute, such as bank8FM and credit rating, the consuming times may depend more on runtime environments than other data sets. Thus, the comparison results may be unconvincing. For the other data sets, i.e., supermarket and hypothyroid, MBTP will take about twenty minutes to generate DICB, whereas ICM only takes half the consuming time of MBTP, i.e., about ten minutes to achieve the same goal.

It should be noted that the consuming time of ICM only contains the updating time from the DICB of decision context to the DICB of granulation context, excluding the consuming time of generating DICB of decision context. Thus, the comparison results above do not imply that ICM is more efficient than MBTP in generating DICB. However, for one thing, since ICM presupposes the presence of DICB of decision context, the comparison is reasonable; for another, since in reality, there may be a continual requirement of attribute granulation, ICM will become more efficient than MBTP because the time of generating DICB of decision context is consumed once and will be balanced finally by the subsequent updating profit.

6 Conclusion and Further Works

In order to reduce the complexity of regenerating decision implications on attribute granulation, this paper stated that the update of decision implications can be accomplished by the update of DICB. Thus, the paper discussed the properties of DICB on attribute granulation and designed an incremental method for updating DICB. Experiment results show that the update of DICB is more efficient than generation from decision context.

It can also be observed that the proposed method has some limitations. For example, the proposed method only considers the refinement of one condition attribute in the update of DICB. Although the updates of multiple attributes can be completed by applying the proposed method to multiple attributes many times, it may be inefficient.

In addition, this paper only examines the update of DICB after condition attribute granulation and does not take decision attribute granulation into consideration. The results in [20,21] show that condition attributes and decision attributes in decision implications are not symmetrical, as reflected in the semantical and syntactical aspects of decision implication. For example, in the syntactical aspect, by AUGMENTATION, one can augment the premises (condition attributes) and reduce the consequences (decision attributes), implying that one cannot apply the proposed method in the paper to the case of decision attribute granulation. Therefore, it is necessary to study the update of decision implications under decision attribute granulation, not only for generating DICB but also for deeply understanding the system of decision implications.

Finally, the proposed method can also be combined with other applications [43]. For example, through attribute refinement, some low-level features in image, such as texture, can be extracted and refined into high-level features, such as texture direction and texture perimeter. By calculating a compact knowledge base such as DICB, features with more semantical information can be obtained. In other words, the information in image can be fully utilized to segment the images, thus improving the accuracy and robustness of segmentation.

Acknowledgement: This work was supported by the National Natural Science Foundation of China (Nos. 61972238, 62072294).

Funding Statement: This research was funded by the National Natural Science Foundation of China (www.nsfc.gov.cn) under Grant No. 61972238 (received by Y. H. Zhai) and No. 62072294 (received by D. Y. Li).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] R. Wille, "Restructuring lattice theory: An approach based on hierarchies of concepts," In: I. Rival (Eds.), *Ordered Sets*, Berlin Heidelberg: Springer, pp. 445–470, 1982.
[2] H. L. Zhi and J. H. Li, "Granule description based on formal concept analysis," *Knowledge-Based Systems*, vol. 104, pp. 62–73, 2016.
[3] Y. H. Zhai, D. Y. Li and K. S. Qu, "Fuzzy decision implications," *Knowledge-Based Systems*, vol. 37, pp. 230–236, 2013.
[4] C. Carpineto and G. Romano, "Rule mining," in *Concept data analysis: Theory and applications*, 1st ed., Hoboken, NJ: Wiley, pp. 155–165, 2004.
[5] Y. H. Zhai, J. J. Qi, D. Y. Li, C. Zhang and W. H. Xu, "The structure theorem of three-way concept lattice," *International Journal of Approximate Reasoning*, vol. 146, no. 12, pp. 157–173, 2022.
[6] X. Y. Chen, J. J. Qi, X. M. Zhu, X. Wang and Z. Wang, "Unlabelled text mining methods based on two extension models of concept lattices," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 2, pp. 475–490, 2020.
[7] A. Houari, W. Ayadi and S. B. Yahia, "A new FCA-based method for identifying biclusters in gene expression data," *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 11, pp. 1879–1893, 2018.
[8] W. H. Xu and W. T. Li, "Granular computing approach to two-way learning based on formal concept analysis in fuzzy datasets," *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 366–379, 2016.
[9] C. F. Zou, H. F. Deng, J. F. Wan, Z. R. Wang and P. Deng, "Mining and updating association rules based on fuzzy concept lattice," *Future Generation Computer Systems*, vol. 82, pp. 698–706, 2018.
[10] A. Mouakher and S. B. Yahia, "On the efficient stability computation for the selection of interesting formal concepts," *Information Sciences*, vol. 472, pp. 15–34, 2019.
[11] C. Carpineto and G. Romano, "A lattice conceptual clustering system and its application to browsing retrieval," *Machine Learning*, vol. 24, no. 2, pp. 95–122, 1996.
[12] W. Y. Bao, C. C. Shen and Y. Li, "Collaborative information retrieval framework based on FCA," in *2015 12th Web Information System and Application Conf. (WISA)*, Jinan, China, pp. 175–178, 2015.
[13] H. L. Zhi, J. J. Qi, T. Qian and R. S. Ren, "Conflict analysis under one-vote veto based on approximate three-way concept lattice," *Information Sciences*, vol. 516, no. 2, pp. 316–330, 2020.
[14] H. L. Zhi, J. H. Li and Y. N. Li, "Multilevel conflict analysis based on fuzzy formal contexts," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 12, pp. 5128–5142, 2022.

[15] G. M. Lang and Y. Y. Yao, "Formal concept analysis perspectives on three-way conflict analysis," *International Journal of Approximate Reasoning*, vol. 152, no. 407, pp. 160–182, 2023.

[16] C. F. Zou, D. Q. Zhang, J. F. Wan, M. M. Hassan and J. Lloret, "Using concept lattice for personalized recommendation system design," *IEEE Systems Journal*, vol. 11, no. 1, pp. 305–314, 2017.

[17] H. Mezni and T. Abdeljaoued, "A cloud services recommendation system based on fuzzy formal concept analysis," *Data Knowledge Engineering*, vol. 116, no. 4, pp. 100–123, 2018.

[18] W. Q. Huang, F. Hao, G. Y. Pang and Y. F. Sun, "Complementary context-enhanced concept lattice aware personalized recommendation," in *2021 IEEE 20th Int. Conf. on Trust, Security and Privacy in Computing and Communications*, Shenyang, China, pp. 919–926, 2021.

[19] K. S. Qu, Y. H. Zhai, J. Y. Liang and M. Chen, "Study of decision implications based on formal concept analysis," *International Journal of General Systems*, vol. 36, no. 2, pp. 147–156, 2007.

[20] Y. H. Zhai, D. Y. Li and K. S. Qu, "Decision implications: A logical point of view," *International Journal of Machine Learning and Cybernetics*, vol. 5, no. 4, pp. 509–516, 2014.

[21] Y. H. Zhai, N. Jia, S. X. Zhang, D. Y. Li and W. H. Xu, "Study on deduction process and inference methods of decision implications," *International Journal of Machine Learning and Cybernetics*, vol. 13, no. 7, pp. 1959–1979, 2022.

[22] Y. H. Zhai, D. Y. Li and K. S. Qu, "Decision implication canonical basis: A logical perspective," *Journal of Computer System Sciences*, vol. 81, no. 1, pp. 208–218, 2015.

[23] D. Y. Li, S. X. Zhang and Y. H. Zhai, "Method for generating decision implication canonical basis based on true premises," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 1, pp. 57–67, 2017.

[24] S. X. Zhang, D. Y. Li and Y. H. Zhai, "Incremental method of generating decision implication canonical basis," *Soft Computing*, vol. 26, no. 3, pp. 1067–1083, 2022.

[25] J. H. Li, C. L. Mei, C. A. Kumar and X. Zhang, "On rule acquisition in decision formal contexts," *International Journal of Machine Learning and Cybernetics*, vol. 4, no. 6, pp. 721–731, 2013.

[26] W. Z. Wu, Y. Leung and J. S. Mi, "Granular computing and knowledge reduction in formal contexts," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 10, pp. 1461–1474, 2009.

[27] K. Y. Qin, B. Li and Z. Pei, "Attribute reduction and rule acquisition of formal decision context based on object (property) oriented concept lattices," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 10, pp. 2837–2850, 2019.

[28] J. P. Xie, M. H. Yang, J. H. Li and Z. Zheng, "Rule acquisition and optimal scale selection in multi-scale formal decision contexts and their applications to smart city," *Future Generation Computer Systems*, vol. 83, no. 3, pp. 564–581, 2018.

[29] H. L. Zhi and Y. F. Ma, "Fuzzy association rule acquisition based on granule description," *Chinese Journal of Kunming University of Science and Technology (Natural Science)*, vol. 47, no. 5, pp. 61–6877, 2022.

[30] Q. Hu and K. Y. Qin, "Attribute reduction and rule acquisition of formal decision context based on dual concept lattice," in *19th Int. Conf. on Web-Based Learning and 5th Int. Symp. on Emerging Technologies for Education*, Ningbo, China, pp. 117–129, 2021.

[31] S. X. Zhang, D. Y. Li, Y. H. Zhai and X. P. Kang, "A comparative study of decision implication, concept rule and granular rule," *Information Sciences*, vol. 508, no. 1, pp. 33–49, 2020.

[32] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems*, vol. 90, no. 2, pp. 111–127, 1997.

[33] X. P. Kang and D. Q. Miao, "A study on information granularity in formal concept analysis based on concept-bases," *Knowledge-Based Systems*, vol. 105, pp. 147–159, 2016.

[34] X. P. Kang, D. Y. Li and S. G. Wang, "Research on domain ontology in different granulations based on concept lattice," *Knowledge-Based Systems*, vol. 27, no. 3, pp. 152–161, 2012.

[35] R. Belohlavek, B. D. Baets and J. Konecny, "Granularity of attributes in formal concept analysis," *Information Sciences*, vol. 260, no. 4, pp. 149–170, 2014.

[36] Y. Wan and L. G. Zou, "An efficient algorithm for decreasing the granularity levels of attributes in formal concept analysis," *IEEE Access*, vol. 7, pp. 11029–11040, 2019.

[37] L. G. Zou, Z. P. Zhang and J. Long, "An efficient algorithm for increasing the granularity levels of attributes in formal concept analysis," *Expert Systems with Applications*, vol. 46, no. 3, pp. 224–235, 2016.
[38] M. W. Shao, M. M. Lv, K. Li and Z. Z. Wang, "The construction of attribute (object)-oriented multi-granularity concept lattices," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 5, pp. 1017–1032, 2020.
[39] L. Wei and Q. Wan, "Granular transformation and irreducible element judgment theory based on pictorial diagrams," *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 380–387, 2016.
[40] J. J. Qi, L. Wei and Q. Wan, "Multi-level granularity in formal concept analysis," *Granular Computing*, vol. 4, no. 3, pp. 351–362, 2019.
[41] Q. Hu and K. Y. Qin, "The construction of multi-granularity concept lattices," *Journal of Intelligent & Fuzzy Systems*, vol. 39, no. 3, pp. 2783–2790, 2020.
[42] L. K. Guo, F. P. Huang, Q. G. Li and G. Q. Zhang, "Power contexts and their concept lattices," *Discrete Mathematics*, vol. 311, no. 18–19, pp. 2049–2063, 2011.
[43] K. Chen, H. Che, M. F. Leung and Y. Wang, "An improved superpixel-based fuzzy C-Means method for complex picture segmentation tasks," in *2022 14th Int. Conf. on Advanced Computational Intelligence (ICACI)*, Wuhan, China, pp. 231–238, 2022.