STRUCTURED ROUGH SET APPROXIMATIONS

A Thesis

Submitted to the Faculty of Graduate Studies and Research

In Partial Fulfilment of the Requirements

For the Degree of

Masters of Science

In

Computer Science

University of Regina

By

Mengjun Hu

Regina, Saskatchewan

October, 2014

UNIVERSITY OF REGINA

FACULTY OF GRADUATE STUDIES AND RESEARCH

SUPERVISORY AND EXAMINING COMMITTEE

Mengjun Hu, candidate for the degree of Master of Science in Computer Science**,** has presented a thesis titled, ***Structured Rough Set Approximations,*** in an oral examination held on October 8, 2014. The following committee members have found the thesis acceptable in form and content, and that the candidate demonstrated satisfactory knowledge of the subject material.

External Examiner:	Dr. Bruce Gilligan, Department of Mathematics & Statistics
Supervisor:	Dr. Yiyu Yao, Department of Computer Science
Committee Member:	Dr. JingTao Yao, Department of Computer Science
Committee Member:	Dr. Cortney J. Butz, Department of Computer Science
Chair of Defense:	Dr. Peter Gu, Faculty of Engineering & Applied Science

ABSTRACT

Rough set theory is widely used in many areas, such as artificial intelligence, machine learning and data mining. Lower and upper approximations are two fundamental notions for concept analysis with rough set theory.

In rough set theory, one can obtain two kinds of sets in an information table, namely, definable and undefinable sets. Intuitively, a definable set represents something we can describe precisely. On the other hand, for an undefinable set, one cannot describe it precisely due to limited available information. One of the main issues in rough set theory is to approximate an undefinable set by a pair of definable sets, called the lower and upper approximations. By introducing these two approximations, approximate inferences can be made about an undefinable set.

There are several formulations of rough set approximations. Pawlak proposed to construct the two approximations as unions of equivalence classes, which is now used in main stream research in rough set theory. By explicitly expressing the individual equivalence classes in Pawlak approximations, Bryniarski used a pair of families of equivalence classes as rough set approximations, which are also known as structured Pawlak approximations. Moreover, Deng et al. proposed the adaptive approximations by using a sequence of equivalence relations with different granularities. Although the latter two formulations take the structure and semantics of the approximations into consideration, they have not received their due attention.

The main objective of this thesis is to present a further exploration of structured and adaptive approximations. We propose a generalized definition of structured rough set approximations from the view of semantics. It can be verified that the proposed structured approximations cover the same sets of objects as Pawlak, Bryniarski and adaptive approximations. In this sense, they are consistent and mathematically equivalent. However, the constituents of these approximations are quite different. The new formulation highlights the semantics of approximations and displays a well-defined internal structure, which will benefit the rule learning process in concept analysis with rough set theory. The comparisons between the proposed structured rough set approximations and Pawlak, Bryniarski, and adaptive approximations are investigated. We also analyze the relationships between the new framework and Grzymała-Busse's LERS systems which are complementary to the new framework.

ACKNOWLEDGEMENT

I would like to extend my sincere gratitude to my supervisor Dr. Yiyu Yao, a professor in computer science at the University of Regina, for his professional guidance and consistent encouragements in my studies and research. His profound knowledge, rigorous scholarship and positive work ethics have greatly impacted me and will continue to do so in my future studies and work.

I deeply appreciate the financial support from my supervisor Dr. Yiyu Yao through his NSERC Discovery Grant, the Faculty of Graduate Studies and Research, and the Department of Computer Science at the University of Regina, which helped to satisfy my daily expenses and kept me concentrated in my studies and research.

Special thanks go to all my colleagues and friends in my office for all the supports and constructive suggestions on my research. In particular, I would like to express my sincere appreciations to my beloved fiancé Xiaofei Deng who has always been providing me with both academic and moral support.

Last but not the least, I would like to thank my whole family, especially my parents for their love and encouragements, and my uncle and aunt who have been supporting me while I live and study in Canada.

POST DEFENSE ACKNOWLEDGEMENT

I would like to thank my internal committee members Dr. Cory Butz and Dr. JingTao Yao, for their constructive and invaluable comments that helped me a lot to improve the thesis. Further appreciation is extended to my external examiner for his insightful suggestions. I would also like to thank the defense chair Dr. Peter Gu for overseeing the defense agenda.

Special thanks to my supervisor Dr. Yiyu Yao and the Faculty of Graduate Studies and Research for all the support to my research and defense.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

This chapter discusses the motivation to introduce the structured rough set approximations. The main contributions of this research are summarized and the thesis structure is outlined.

1.1 Motivation

According to the classical view of concepts, a concept is represented by a pair of intension and extension. Specifically, an intension consists of properties describing the concept, and an extension consists of objects that are instances of the concept [34,43]. In rough set theory [26], a description language [17], or, equivalently, families of sets of attribute-value pairs [9,10], is introduced to formally represent intensions of concepts, and sets of objects are used to represent extensions of concepts. For an arbitrary set of objects representing the extension of a concept, its intension may not be able to be formally represented due to limited information. Intuitively, the concepts whose intensions can be precisely represented are called definable concepts, while others are called undefinable concepts. Correspondingly, the extension of a definable concept is called a definable set, while the extension of an undefinable concept is called an undefinable set.

A fundamental construct of rough set theory is the approximation of an undefinable set by a pair of definable sets called the lower and upper approximations [4, 24–26, 38]. At least three notions of the definability of sets appear in rough set literature. In one notion, a set is said to be definable, originally called describable by Marek and Pawlak [17], if the set is exactly the set of objects satisfying a formula in a description language, that is, when we can use a logic formula to describe the set. In the second notion, a set is said to be definable if it is the union of a family of equivalence classes induced by an equivalence relation [25], where the family is a subset of the partition of the equivalence relation. Finally, a set is said to be definable if its lower and upper approximations are the same as the set itself. Although the three definitions of definability are mathematically equivalent, they are very different semantically. The first two definitions treat definability as a primitive notion that motivates the introduction of rough set approximations. In other words, approximations are needed in order to make inferences about undefinable sets [26]. In comparison, the description-language-based definition is advantageous over equivalence-relation-based definition, because the latter does not provide a satisfactory answer to the question: why is the union of a family of equivalence classes a definable set? The third definition treats approximations as primitive notions and definability as a derived notion. This definition fails to explain the reason for introducing rough set approximations. By taking advantages of the semantically superior description-language-based definition, we propose the notion of structured rough set approximations.

According to Pawlak [24], for a given set, its lower approximation is defined as the greatest definable set contained by it and its upper approximation is the least definable set containing it. There are many formulations of rough set approximations. Pawlak approximations [24] based on an equivalence relation are used in the main stream research.

Definition 1. *Let $E \subseteq U \times U$ denote an equivalence relation (i.e., a reflexive, sym-*

metric and transitive binary relation) on a nonempty and finite set U of objects. The equivalence class containing x is given by:

$$[x] = \{y \in U \mid xEy\}. \tag{1.1}$$

All the equivalence classes induced by E form a partition U/E of U, that is, a family of nonempty and pairwise disjoint subsets of U whose union is U:

$$U/E = \{[x] \mid x \in U\}. \tag{1.2}$$

With respect to an information table, equivalence classes are interpreted as minimal nonempty definable sets called elementary sets [38]. The partition U/E induced by the equivalence relation E consists of all elementary sets that are the building blocks to construct Pawlak approximations.

Definition 2. *For a subset of objects $X \subseteq U$, Pawlak lower and upper approximations [24] are defined through equivalence classes as follows:*

$$\begin{aligned}
\underline{apr}(X) &= \bigcup\{[x] \in U/E \mid [x] \subseteq X\} \\
&= \{x \in U \mid [x] \subseteq X\}, \\
\overline{apr}(X) &= \bigcup\{[x] \in U/E \mid [x] \cap X \neq \emptyset\} \\
&= \{x \in U \mid [x] \cap X \neq \emptyset\}.
\end{aligned} \tag{1.3}$$

By definition, both Pawlak lower and upper approximations are subsets of U. Bryniarski [3] proposed an alternative definition by removing the union in Pawlak approximations.

Definition 3. *For a subset of objects $X \subseteq U$, Bryniarski [3] lower and upper approx-*

imations are defined as follows:

$$
\begin{aligned}
\underline{bapr}(X) &= \{[x] \in U/E \mid [x] \subseteq X\}, \\
\overline{bapr}(X) &= \{[x] \in U/E \mid [x] \cap X \neq \emptyset\}.
\end{aligned}
\tag{1.4}
$$

By definition, Bryniarski lower and upper approximations are no longer subsets of objects, but are families of equivalence classes, namely, $\underline{bapr}(X) \subseteq U/E$ and $\overline{bapr}(X) \subseteq U/E$. While the Pawlak definition is widely used in the main stream research, only a few studies [7] consider the Bryniarski definition.

Theorem 1. *Pawlak and Bryniarski approximations can define each other as follows:*

$$
\begin{aligned}
\underline{apr}(X) &= \bigcup \underline{bapr}(X), \\
\overline{apr}(X) &= \bigcup \overline{bapr}(X),
\end{aligned}
\tag{1.5}
$$

and

$$
\begin{aligned}
\underline{bapr}(X) &= \{[x] \in U/E \mid [x] \subseteq \underline{apr}(X)\}, \\
\overline{bapr}(X) &= \{[x] \in U/E \mid [x] \subseteq \overline{apr}(X)\}.
\end{aligned}
\tag{1.6}
$$

The proof is given in Section A.1 of Appendix A.

Although the two definitions differ slightly in form, they are very different semantically. By using the equivalence classes in the lower and upper approximations, $\underline{bapr}(X)$ and $\overline{bapr}(X)$ preserve the structural information about rough set approximations, which was missing from the Pawlak definition.

There are several reasons to keep the structural information. When we construct rules from an approximation, we use each equivalence class to form one classification rule. An explicit representation of the structural information makes it much easier to explain and understand such a learning task. Each equivalence class may be viewed as a granule and a structured approximation explicitly shows the composition of a family of granules in forming a rough set approximation. This interpretation connects rough set theory and granular computing [37].

Pawlak approximations use only one equivalence relation. Deng et al. [6] proposed using a set of equivalence relations with different granularities and construct the adaptive approximations. By adding more attributes, equivalence classes induced by attributes will be finer, that is, with fewer objects. To construct the adaptive approximations of a given set, equivalence classes, from coarser to finer, are used sequentially. As a result, we may use large equivalence classes, which can create general classification rules. Moreover, a hierarchical structure is induced in the sequence of approximations.

This thesis proposes a generalized definition of structured rough set approximations by considering the family of all conjunctively definable sets, of which the partition U/E is a subset. This new formulation of approximations will put emphasis on the semantics of approximations. It is useful in the following aspects:

(1) learning process of rules;

(2) interpretation of rules;

(3) examination of the structure of a set of rules.

This new formulation of approximations can benefit concept analysis with rough set theory in terms of creating more general or simple classification rules, which is an essential task of rough set theory. In a real world application, attributes in classification rules are usually associated with certain kinds of cost. For instance, in medical diagnosis, a symptom may be represented by an attribute and the corresponding cost may be obtained from several related medical tests. In order to reduce costs and learn simple rules, large equivalence classes in an approximation are preferred. However, most formulations of approximations may use small equivalence classes and pay little attention to the semantics of the components.

In Pawlak and adaptive approximations, equivalence classes are pairwise disjoint and there do not exist redundant equivalence classes. In the structured approximations proposed in this study, there may exist redundant conjunctively definable sets. It is therefore necessary to remove the redundancy. The notion of a union-reduct or $\cup$-reduct is adopted and two methods for deriving reduced structured approximations are introduced. A $\cup$-reduct of a family of sets of objects is a minimal subset of the family that covers the same objects as the family [26].

1.2 Contributions of this Research

The six main contributions of the research presented in this thesis can be summarized as follows.

1. A generalized definition of structured rough set approximations is proposed. Two methods to reduce redundancy in a structured lower approximation are discussed.

2. Relationships between structured rough set approximations, Pawlak approximations and adaptive approximations are investigated. Relationships between structured rough set approximations and a covering-based system LERS proposed by Grzymała-Busse [9, 10] are discussed.

3. While many formulations of rough set approximations do not consider much about semantics, structured rough set approximations focus on a well-defined structure that explicitly provides semantics of approximations.

4. With a well-defined structure, structured approximations are useful for concept analysis with rough set theory. In other words, simpler rules may be learned from structured approximations than from other formulations.

5. Structured approximations are based on a covering-based formulation of rough set approximations, which generalizes partition-based approximations such as Pawlak and adaptive approximations. A covering is a family of nonempty subsets of the universe (i.e., all the objects) whose union is the universe. Different from a partition in which the components must not have overlap, a covering allows overlap between its components.

6. Although covering-based rough set models have been discussed by many researchers, a semantical interpretation of patches or components in a covering is still not well defined. Since each component in a structured approximation has a direct semantical interpretation, structured approximations will generate a semantically sound model for covering-based rough set models.

1.3 Thesis Structure

A review of the basic ideas and notions of concept analysis with rough set theory based on an information table is given in Chapter 2. Most importantly, the notion of definability of sets and rough set approximations are examined. Chapter 3 gives an analysis of two types of approximations with structural information, that is, structured Pawlak approximations and adaptive approximations. A definition of the proposed structured rough set approximations is introduced in Chapter 4. Moreover, its relationships to Pawlak approximations, adaptive approximations and LERS systems are investigated and two methods to reduce redundancy in the structured lower approximations are suggested. Chapter 5 concludes the thesis and discusses possible future work related to the research presented in this thesis. The proofs of all theorems in this thesis are given in the Appendix A.

Chapter 2

CONCEPT ANALYSIS BASED ON AN INFORMATION TABLE

This chapter provides a review of basic notions and ideas about concept analysis with rough set theory. Most notions and ideas are from Pawlak rough set model [24, 26] which forms the foundation of rough set theory. In addition, we give an extended exploration of these notions and ideas, which provides background knowledge for our research.

2.1 Interpretations of Concepts

Concept analysis is a main task in rough set theory. The notion of concepts plays a basic role in rough sets. This section reviews theories related to concept interpretation.

2.1.1 The Classical View of Concepts

Generally, a concept can be viewed as a fundamental classification denoting a group of objects [5, 21]. The learning and analysis of concept is involved in many disciplines, including artificial intelligence, machine learning, information retrieval, phycology, philosophy, cognitive science, data analysis and mathematics [20, 31, 32, 34, 43].

There are many views and theories about the formulation of concepts, such as the classical view, the prototype view, the frame view and the theory view [5, 21, 34]. Rough set theory takes the classical view of concepts, in which a concept is constituted by a pair of intension and extension [39]. Intension is a set of all the attributes, properties or features of the concept and can be used to figure out whether an object belongs to the concept. In other words, all objects belonging to a concept must satisfy all the properties described by the intension of the concept. Generally speaking, intension takes the form of the definition of a concept. Extension comprises all the objects belonging to the concept. That is, the objects satisfying all the properties described by the corresponding intension. Extension can be represented by a set of objects satisfying certain conditions or having certain properties. A pair of intension and extension describes two aspects of a concept, with intension indicating the internal content of a concept and extension indicating the particular objects that the concept denotes.

To some extent, all the views about concepts are developments of, or reactions to, the classical view of concepts. In the last thirty years, there have been some criticisms about the limitations of the classical view of concepts [18]. In spite of those criticisms, the classical view of concepts is sufficient for and applicable to our research in rough set theory.

2.1.2 Interpreting Concepts by Meaning Triangle

The meaning triangle, also known as the triangle of reference and the semiotic triangle, was first proposed by Ogden and Richards [22] in 1923 to model the relationship between a symbol and the objects it represents. The meaning triangle has been used as a tool for organizing information in the fields of concepts-oriented knowledge acquisition, texts analysis, semiotic analysis and so on [8, 19, 28, 32]. Figure 2.1 illustrates the meaning triangle proposed by Ogden and Richards. This triangle describes the relationships between a unit of thought, a symbol (e.g. a name) and the referent (e.g. a hieroglyph) [22].

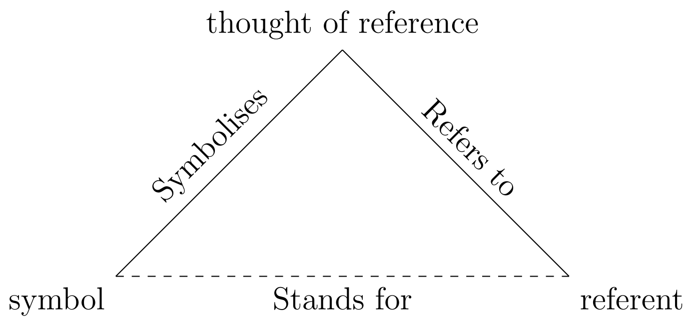


Figure 2.1: Ogden and Richards' Meaning Triangle

According to the classical view of concepts, there are two aspects in understanding a concept, namely, the intension and extension. From the viewpoint of the meaning triangle, intension can serve as thought of reference and extension as referent. Symbol in a meaning triangle is actually the name of a concept. Hence, by applying the meaning triangle, a concept can be represented by Figure 2.2 [39]. In Figure 2.2, name can be considered as a label which symbolises the concept. From the intension, we can get references to the corresponding extension.

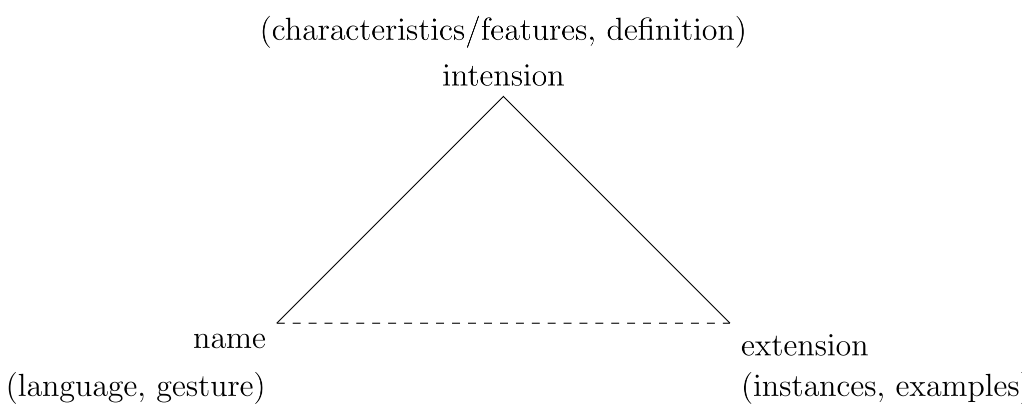


Figure 2.2: Meaning Triangle of Concepts

In rough set theory, one can establish two formulations based on the meaning triangle of concepts. In the first formulation, name is considered as a sign taking the form of natural language; intension is considered as a set of properties and extension is considered as a set of instances. Hence, both intension and extension can be operated according to set theory. This view is explicitly expressed by Figure 2.3. The second formulation, which is given by Figure 2.4, adopts a more operational form, that is, logic formulas are used to express the intension. As a result, a relationship between rough set theory and logic can be established.

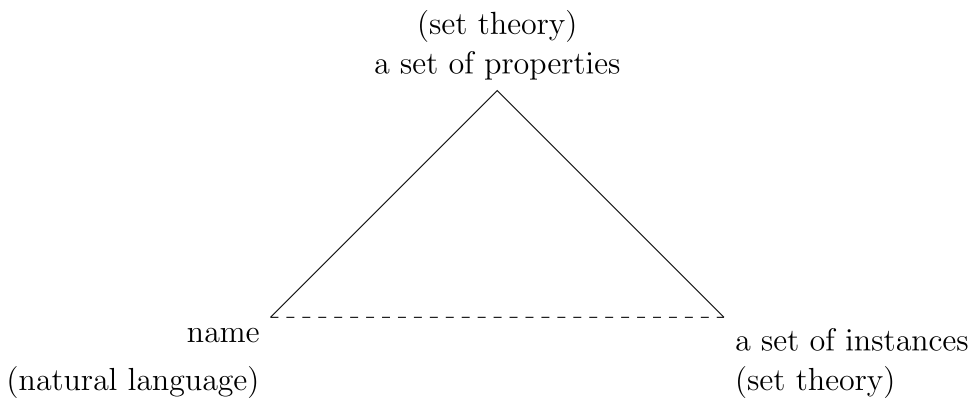


Figure 2.3: The First Formulation of Meaning Triangle of Concepts

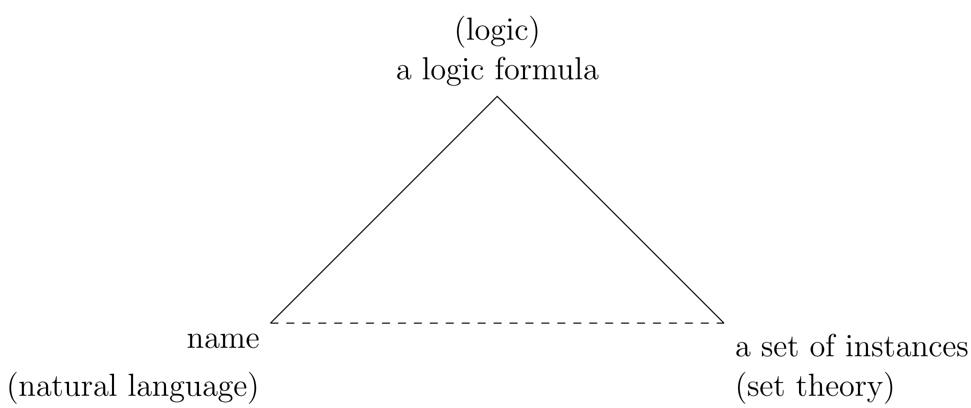
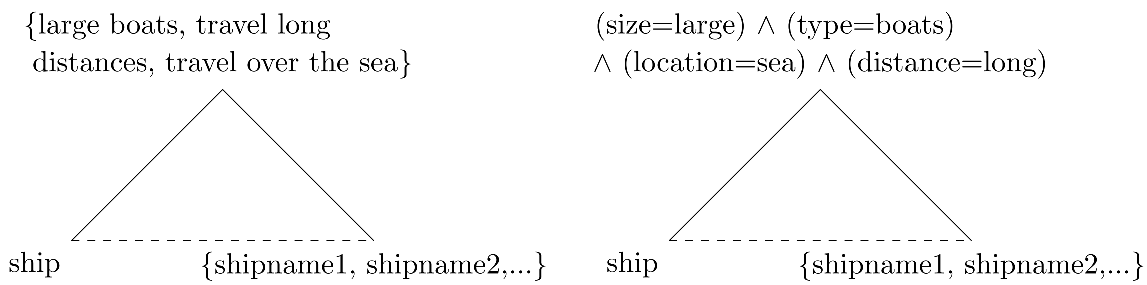


Figure 2.4: The Second Formulation of Meaning Triangle of Concepts

In the following, a concrete example of the concept named “ship” to better illustrate the idea. The definition of “ship” given by Merriam-Webster dictionary is “a large boat used for traveling long distances over the sea.”[1] This can be the intension of the concept “ship.” For its extension, each individual ship is included and labeled by its name. Based on these facts, the two formulations of concept “ship” are given by Figure 2.5. In the second formulation, the operator $\wedge$ contained in a formula is logic conjunction.



(a) The first formulation

(b) The second formulation

Figure 2.5: Two Formulations of Meaning Triangle of Concept “ship”

The meaning triangle shows a complete vision of a concept based on the classical view. In rough set theory, the classical view and meaning triangle of a concept are adopted to formulate a concept.

2.2 Representing Concepts Using an Information Table

Based on the classical view of concepts, a concept can be represented as a pair of intension and extension. In rough set theory, intension and extension are constructed with respect to an information table.

[1] URL: http://www.merriam-webster.com/dictionary/ship.

2.2.1 An Information Table

In rough set theory, an information table is used to represent information or data of a set of objects. An information table is originally called an information system by Pawlak [25]. The term "information table" avoids confusions with other commonly associated meanings of an information system. An information table contains all the available data in a tabular form with each row representing an object, each column representing an attribute and each cell representing the value of an object on an attribute.

Definition 4. *An information table is defined as a tuple:*

$$T = (U, AT, \{V_a \mid a \in AT\}, \{I_a \mid a \in AT\}), \tag{2.1}$$

where U *is a finite nonempty set of objects called the universe;* AT *is a finite nonempty set of attributes;* V_a *is the domain of attribute* a*; and* $I_a : U \to V_a$ *is a description function that assigns a value from* V_a *to each object.*

By definition, $I_a(x) \in V_a$ denotes the value of an object x on attribute a.

Example 1. *Table 2.1 is an example of an information table revised from the one used in [27, 39]. With respect to the definition of an information table, in Table 2.1, we have:*

- *The universe is* $U = \{o_1, o_2, o_3, o_4, o_5, o_6\}$*;*
- *The set of attributes is* $AT = \{\text{Height, Hair, Eyes}\}$*;*
- *The set of the domains of the four attributes is* $\{V_{\text{Height}}, V_{\text{Hair}}, V_{\text{Eyes}}\} = \big\{\{\text{short, tall}\}, \{\text{blond, red}\}, \{\text{blue, brown}\}\big\}$*;*
- *The description function is described by all the cells in the information table, such as* $I_{\text{Height}}(o_1) = \text{short}$*, which means object* o_1 *takes value* short *on attribute*

Height.

Object	Height	Hair	Eyes
o_1	short	blond	blue
o_2	short	blond	brown
o_3	short	red	blue
o_4	tall	blond	blue
o_5	tall	red	brown
o_6	tall	red	brown

Table 2.1: An Example of an Information Table

Each row describes all the considered attributes or properties of a specific object and such properties of objects are perceived through assigning values to the attributes.

This study only considers the case where each object takes exactly one value on each attribute. In situations where it is impossible to get the exact values of certain attributes, incomplete information tables are used and investigated [4, 14, 46, 47].

2.2.2 Representing Concepts with a Description Language

Based on an information table, both the intension and the extension of a concept can be formally represented. Specifically, extension is represented as a subset of objects. For intension, there can be two different kinds of representation. One representation uses formulas defined by a description language and the other one equivalently uses a family of sets of attribute-value pairs. This section focuses on the representation with a description language.

A sublanguage of the description language introduced by Marek and Pawlak [17] is adopted, which considers only logic conjunction and disjunction. These two connectives are sufficient for this study.

Definition 5. *Formulas in the description language DL is defined as follows:*

(1) Atomic formulas:

$$(a = v) \in DL, \textit{ where } a \in AT \textit{ and } v \in V_a.$$

(2) Composite formulas:

$$\textit{If } p, q \in DL, \textit{ then } p \wedge q \in DL \textit{ and } p \vee q \in DL.$$

If a formula in DL contains only conjunction, it is called a conjunctive formula.

Example 2. *In Table 2.1,*

(1) an example of atomic formulas is:

$$\text{Height} = \text{short};$$

(2) an example of composite formulas is:

$$\big((\text{Height} = \text{short}) \wedge (\text{Hair} = \text{red})\big) \vee (\text{Hair} = \text{blond}).$$

The description language DL is closely related to description logic [1]. An atomic formula $a = v$ expresses a condition which can be described as taking value v on attribute a. In the description language DL, we explicitly give the formulas for defining concepts. With respect to an information table where there is a finite set of attributes, the two logic connectives are sufficient for the discussion. In fact, they provide two ways to construct concepts. That is, more general concepts can be built by taking disjunction and more specific concepts can be built by taking conjunction.

A formula in DL is interpreted by a subset of objects satisfying the formula.

Definition 6. *Let $x \models p$ denote that an object $x \in U$ satisfies a formula $p \in DL$. The satisfiability is given as:*

$$\begin{aligned}
&(1) \qquad x \models (a = v), \ \text{if} \ I_a(x) = v, \\
&(2) \qquad x \models p \wedge q, \ \text{if} \ x \models p \text{ and } x \models q, \\
&(3) \qquad x \models p \vee q, \ \text{if} \ x \models p \text{ or } x \models q.
\end{aligned} \tag{2.2}$$

Example 3. *In Table 2.1, we have:*

$$\begin{aligned}
&(1) \qquad o_1 \models (\text{Height} = \text{short}), \\
&(2) \qquad o_2 \models (\text{Hair} = \text{blond}) \wedge (\text{Eyes} = \text{brown}), \\
&(3) \qquad o_3 \models (\text{Height} = \text{tall}) \vee (\text{Hair} = \text{red}).
\end{aligned}$$

Based on the notion of satisfiability, one can construct a meaning set of a formula by collecting all objects satisfying the formula.

Definition 7. *Given a formula $p \in DL$, its meaning set is a subset of U defined by:*

$$m(p) = \{x \in U \mid x \models p\}. \tag{2.3}$$

Example 4. *In Table 2.1, we have* $m(\text{Height} = \text{short} \wedge \text{Hair} = \text{blond}) = \{o_1, o_2\}$.

By definition, one can establish the following connections between logic connectives and set-theoretic operators through the notion of a meaning set [24–26].

Theorem 2. *The meaning set of a formula in DL can be computed as:*

$$\begin{aligned}
&(1) \qquad m(a = v) = \{x \in U \mid I_a(x) = v\}, \\
&(2) \qquad m(p \wedge q) = m(p) \cap m(q), \\
&(3) \qquad m(p \vee q) = m(p) \cup m(q).
\end{aligned} \tag{2.4}$$

That is, the logic conjunction is interpreted by set intersection and disjunction by set union.

The proof is given in Section A.2 of Appendix A.

By using the description language, the intension of a concept may be represented by a formula and its extension may be represented by the meaning set of the formula. For two concepts, $(p, m(p))$ and $(q, m(q))$, we can construct two new concepts $(p \wedge q, m(p) \cap m(q))$ and $(p \vee q, m(p) \cup m(q))$.

2.2.3 Representing Concepts with Attribute-Value Pairs

Another approach to formulate intension is by using attribute-value pairs. This kind of representation can be equivalently converted into the form with formulas in DL.

Basically, an attribute-value pair is denoted by (a, v) with $a \in AT$ and $v \in V_a$, which expresses a condition that an object takes value v on attribute a. Based on an information table, the set of all possible attribute-value pairs is denoted as:

$$AV = \{(a, v) \mid a \in AT, v \in V_a\}. \tag{2.5}$$

For representing an intension of a concept, a family of sets of attribute-value pairs is used, which can be represented as:

$$\{S_1, S_2, \cdots, S_n\}, \tag{2.6}$$

where $S_i \subseteq AV (1 \leq i \leq n)$.

Definition 8. *The satisfiability of attribute-value pairs is defined as follows:*

(1) for an attribute-value pair (a, v):

$$x \models (a, v), \text{ if } I_a(x) = v; \tag{2.7}$$

(2) for a set of attribute-value pairs $S \subseteq AV$:

$$x \models S, \text{ if } \forall (a,v) \in S, x \models (a,v); \tag{2.8}$$

(3) for a family of sets of attribute-value pairs $\{S_1, S_2, \cdots, S_n\}(S_i \subseteq AV, 1 \leq i \leq n)$:

$$x \models \{S_1, S_2, \cdots, S_n\}, \text{ if } \exists S_i (1 \leq i \leq n), x \models S_i. \tag{2.9}$$

Example 5. *In Table 2.1, we have:*

$$\begin{array}{ll} (1) & o_1 \models (\text{Height}, \text{short}), \\ (2) & o_2 \models \{(\text{Hair}, \text{blond}), (\text{Eyes}, \text{brown})\}, \\ (3) & o_3 \models \big\{\{(\text{Height}, \text{tall})\}, \{(\text{Hair}, \text{red})\}\big\}. \end{array}$$

Based on the satisfiability of attribute-value pairs, one can construct the meaning set of a family of sets of attribute-value pairs.

Definition 9. *Suppose $\{S_1, S_2, \cdots, S_n\}$ is a family of sets of attribute-value pairs where $S_i \subseteq AV(1 \leq i \leq n)$. Its meaning set is defined as:*

$$m(\{S_1, S_2, \cdots, S_n\}) = \{x \in U \mid x \models \{S_1, S_2, \cdots, S_n\}\}. \tag{2.10}$$

Theorem 3. *Suppose $\{S_1, S_2, \cdots, S_n\}$ is a family of sets of attribute-value pairs where $S_i \subseteq AV(1 \leq i \leq n)$. One can compute its meaning set by using set intersection and set union as:*

$$\begin{aligned} m(\{S_1, S_2, \cdots, S_n\}) &= \bigcup_{1 \leq i \leq n} m(\{S_i\}) \\ &= \bigcup_{1 \leq i \leq n} \bigcap_{(a,v) \in S_i} \{x \mid x \in U, x \models (a,v)\}. \end{aligned} \tag{2.11}$$

The proof is given in Section A.3 of Appendix A.

A family of sets of attribute-value pairs is used to represent the intension of a concept. In the case that the family is a singleton set $\{S\}$ where $S \subseteq AV$, S is briefly used to denote it. In the case that the set S is a singleton set $\{(a, v)\}$, (a, v) is briefly used instead.

Families of sets of attribute-value pairs and formulas in DL can be converted to each other. Attribute-value pairs in one set have a kind of "and" relationship, while sets of attribute-value pairs in one family have a kind of "or" relationship. For a family of sets of attribute-value pairs $\{S_1, S_2, \cdots, S_n\}$ where $S_i \subseteq AV(1 \leq i \leq n)$, one can convert it into a formula in DL following three steps:

Step 1: For each $S_i(1 \leq i \leq n)$, convert each attribute-value pair $(a, v) \in S_i$ into an atomic formula $a = v$;

Step 2: Convert $S_i(1 \leq i \leq n)$ into a conjunctive formula by taking conjunction between the atomic formulas constructed from S_i;

Step 3: Take disjunction between all conjunctive formulas constructed in Step 2 and get the formula corresponds to the family $\{S_1, S_2, \cdots, S_n\}$.

A formula in DL can always be converted to a disjunctive normal formula (DNF) [11] which is a form of disjunction of conjunctive formulas. Suppose f is a formula in DL. Its DNF is denoted as $f_1 \vee f_2 \vee \cdots \vee f_n$ where $f_i(1 \leq i \leq n)$ is a conjunctive formula. One can convert f into a family of sets of attribute-value pairs following three steps:

Step 1: For each $f_i(1 \leq i \leq n)$, convert each atomic formula $a = v$ in f_i into an attribute-value pair (a, v);

Step 2: Convert $f_i(1 \leq i \leq n)$ into a set of attribute-value pairs constructed in Step 1;

Step 3: Put all the n sets of attribute-value pairs constructed in Step 2 into a family which is the family of sets of attribute-value pairs corresponding to formula f.

2.3 A Definition and Categorization of Definable Sets

This section reviews the notion of the definability of sets and three categories of definable sets. All discussion in this section is based on the representation of concepts with the description language DL. One can equivalently discuss all the notions from the view of attribute-value pairs.

2.3.1 Definable and Undefinable Sets

In the Pawlak rough set model, all sets of objects can be classified into two groups, that is, the definable and undefinable sets. Intuitively, based on an information table, a definable set can be described and defined precisely, while an undefinable set cannot.

Definition 10. *A subset $X \subseteq U$ is a definable set if there exists a formula $p \in DL$ such that*

$$X = m(p). \tag{2.12}$$

Otherwise, X is undefinable.

By definition, a definable set X is the meaning set of a formula p, and the formula p is a description of objects in X.

Example 6. *In Table 2.1, $\{o_1, o_2\}$ is a definable set since $\{o_1, o_2\} = m\big((\text{Height} = \text{short}) \wedge (\text{Hair} = \text{blond})\big)$. However, $\{o_6\}$ is an undefinable set since there does not exist a formula describing the set $\{o_6\}$. This is because o_5 and o_6 take the same values on all* *attributes. No formula can differentiate them and any set including only one of them will be undefinable.*

Marek and Pawlak [17] called a definable set a describable set. A formula p of a definable set X enables us to tell if an object is in X or not. The definition of a definable set only requires the existence of one formula. In many cases, such a formula is not unique. One may find more than one formula that defines the same set of objects.

The definability of a subset of objects can reflect the definability of the corresponding concept. For a definable set $X = m(p)$, we can form a concept $(p, m(p))$ with p representing the intension and X representing the extension. Such a concept is called a definable concept. If the subset $Y \subseteq U$ is an undefinable set, the concept with Y as its extension is considered to be undefinable. The following discussion focuses on definability of sets and one can investigate the definability of concepts similarly.

Let DEF (U) denote the family of all definable sets on the universe U. The family of all undefinable sets is given by $2^U - \mathrm{DEF}(U)$ where 2^U denotes the power set of U, that is, the family of all subsets of U. Marek and Pawlak [17] proposed the following theorem on the structure of $\mathrm{DEF}(U)$.

Theorem 4. $\mathrm{DEF}(U)$ *is a Boolean algebra. That is,* $\mathrm{DEF}(U)$ *includes the empty set* $\emptyset$ *and the universe* U*, and it is closed under set complement* ^{c}*, intersection* $\cap$ *and union* $\cup$*.*

Since the description language DL used in this study is slightly different from the one used by Marek and Pawlak, Theorem 4 is verified in Section A.4 of Appendix A.

2.3.2 Three Categories of Definable Sets

Based on the different forms of formulas, we can obtain three particular categories of definable sets. These three categories play important roles in rough set theory.

Definable Basic Sets

The definable basic sets are defined by atomic formulas. Formally, a definable basic set can be represented as:

$$X = m(a = v), \tag{2.13}$$

where $a \in AT$ and $v \in V_a$.

Example 7. *In Table 2.1,* $\{o_1, o_2, o_3\} = m(\text{Height} = \text{short})$ *is a definable basic set.*

A definable set is considered as basic if its formula has the basic form (i.e., atomic formulas).

Conjuntively Definable Sets

If a set can be defined by a conjunctive formula, it is called a conjunctively definable set [45]. That is, a conjunctively definable set $X \subseteq U$ can be represented as:

$$X = m\big((a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_m = v_m)\big), \tag{2.14}$$

where $a_i \in AT$ and $v_i \in V_{a_i}$ $(1 \leq i \leq m)$.

Example 8. *In Table 2.1,* $\{o_1, o_2, o_3\} = m(\text{Height} = \text{short})$ *and* $\{o_1, o_4\} = m\big((\text{Hair} = \text{blond}) \wedge (\text{Eyes} = \text{blue})\big)$ *are both conjunctively definable sets.*

Let CDEF(U) denote the family of all conjunctively definable sets. It can be easily verified that CDEF(U) is closed under set intersection $\cap$ and that CDEF(U) $\subseteq$ DEF(U). Definable basic sets are actually the maximal conjunctively definable sets. Furthermore, a conjunctively definable set can be expressed as the intersection of a family of definable basic sets.

Definable Elementary Sets

Consider a special conjunctive formula which is formed by conjunction of exactly one atomic formula from each attribute. Suppose the set of all attributes is $AT = \{a_1, a_2, \cdots, a_n\}$. Such a formula can be represented as follows:

$$(a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_n = v_n), \tag{2.15}$$

where $a_i \in AT$, $v_i \in V_{a_i}$, and $a_i \neq a_j$ when $i \neq j$ $(1 \leq i, j \leq n)$. Such formulas are called minterms.

A subset $X \subseteq U$ is called a definable elementary set if there exists a minterm t such that $X = m(t)$. A definable elementary set is a minimal nonempty definable set.

Example 9. *In Table 2.1,* $\{o_1\} = m\big((\text{Height} = \text{short}) \wedge (\text{Hair} = \text{blond}) \wedge (\text{Eyes} = \text{blue})\big)$ *is a definable elementary set.*

2.3.3 Properties of the Three Categories of Definable Sets

By definitions, we can obtain four properties of the relationships between the three categories of definable sets as follows.

(1) The first property is that definable basic sets and definable elementary sets are special cases of conjunctively definable sets.

    It can be easily verified using definitions. In addition, a definable basic set is a maximal conjunctively definable set, and a definable elementary set is a minimal one.

(2) The second property is that a conjunctively definable set can be interpreted as the intersection of a family of definable basic sets.

    Suppose a conjunctively definable set X is defined by formula $(a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_m = v_m)$. The set X can be equivalently expressed by the intersection of definable basic sets as follows:

$$
\begin{aligned}
X &= m\big((a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_m = v_m)\big) \\
&= \bigcap_{1 \leq i \leq m} m(a_i = v_i). \qquad (2.16)
\end{aligned}
$$

(3) The third property is that a conjunctively definable set can be interpreted as the union of a family of definable elementary sets.

If an attribute a appears more than once, say twice, in a conjunctive formula, that is, both $(a = v_1)$ and $(a = v_2)$ appear in the formula, the appearance of a can actually be reduced to once. In the case that $v_1 = v_2$, we can preserve only one atomic formula of a, while in the case that $v_1 \neq v_2$, the formula will lead to the empty set, which can be viewed as the union of an empty family of definable elementary set. Thus, we suppose that a conjunctively definable set X is defined by formula $p = (a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_m = v_m)$ where each attribute appears at most once. We verify that p can be equivalently transformed into disjunction of minterms by expanding p with attributes which do not appear in p. Let A_p denote the set of attributes that appear in p, that is, $A_p = \{a_1, a_2, \cdots, a_m\}$. Its complement set A_p^c contains the attributes that do not appear in p. In order to include an attribute $a_c \in A_p^c$ in p, we replace p with the following formula:

$$p \wedge \big(\bigvee_{v \in V_{a_c}} a_c = v \big) \qquad (2.17)$$

which has the same meaning set with p:

$$\begin{aligned} m\big(p \wedge (\bigvee_{v \in V_{a_c}} a_c = v)\big) &= m(p) \cap m(\bigvee_{v \in V_{a_c}} a_c = v) \\ &= m(p) \cap \big(\bigcup_{v \in V_{a_c}} m(a_c = v)\big) \\ &= m(p) \cap U \\ &= m(p). \end{aligned} \tag{2.18}$$

By noticing that:

$$\begin{aligned} m\big(p \wedge (\bigvee_{v \in V_{a_c}} a_c = v)\big) &= m\big(\bigvee_{v \in V_{a_c}} (p \wedge (a_c = v))\big) \\ &= \bigcup_{v \in V_{a_c}} m\big(p \wedge (a_c = v)\big), \end{aligned} \tag{2.19}$$

it follows that:

$$\begin{aligned} X &= m(p) \\ &= m\big(p \wedge (\bigvee_{v \in V_{a_c}} a_c = v)\big) \\ &= \bigcup_{v \in V_{a_c}} m\big(p \wedge (a_c = v)\big), \end{aligned} \tag{2.20}$$

which means X can be expressed as the union of a family of all conjunctively definable sets whose formulas contain one more attribute than p. By adding all attributes in A_p^c recursively, one can finally express X as the union of a family of definable elementary sets.

(4) The fourth property is that the set of all definable sets, that is, DEF(U), is an atomic Boolean algebra.

From Theorem 4 in Section 2.3.1, we know that DEF(U) is a Boolean algebra. By noticing that definable elementary sets are the minimal nonempty definable sets, we only need to show that any definable set $X = m(p)$ can be represented as union of definable elementary sets. Mathematically, we know that p can be equivalently represented in the following form called disjunctive normal form (DNF) [11]:

$$p = p_1 \vee p_2 \vee \cdots \vee p_n, \tag{2.21}$$

where each $p_i (1 \leq i \leq n)$ is a conjunctive formula. Then it follows that:

$$\begin{aligned} X &= m(p) \\ &= m(p_1 \vee p_2 \vee \cdots \vee p_n) \\ &= \bigcup_{1 \leq i \leq n} m(p_i), \end{aligned} \tag{2.22}$$

where each $m(p_i)(1 \leq i \leq n)$ is a conjunctively definable set. By the third property, each conjunctively definable set $m(p_i)$ can be represented as the union of a family of definable elementary sets. Consequently, X can be represented as the union of a family of definable elementary sets. Thus, DEF(U) is an atomic Boolean algebra with definable elementary sets as atoms.

2.3.4 P-definable Sets

The notions and ideas concerning definability can be extended by considering a subset of attributes instead of AT.

Definition 11. *Given a set of attributes $P \subseteq AT$, a subset $X \subseteq U$ is a P-definable set if there exists a formula $p_P \in DL$ and p_P contains only attributes from P such that*

$$X = m(p_P). \tag{2.23}$$

Otherwise, X is P-undefinable.

Example 10. *In Table 2.1,* $\{o_1, o_2\}$ *is* {Height, Hair}*-definable set since* $\{o_1, o_2\} = m(\text{Height} = \text{short} \wedge \text{Hair} = \text{blond})$*. However,* $\{o_1, o_2\}$ *is* {Height}*-undefinable since there does not exist a formula which contains only attribute* Height *and defines* $\{o_1, o_2\}$*.*

The family of all P-definable sets is denoted as $\text{DEF}_P(U)$. By definitions, definable sets discussed previously are actually AT-definable sets, that is, $\text{DEF}(U) = \text{DEF}_{AT}(U)$. For a single attribute $a \in AT$, we use $\{a\}$-definable sets and a-definable sets interchangeably.

For two subsets of attributes, $P, Q \subseteq AT$, we have:

$$P \subseteq Q \implies \text{DEF}_P(U) \subseteq \text{DEF}_Q(U) \subseteq \text{DEF}_{AT}(U). \tag{2.24}$$

That is, the families of definable sets are monotonous with respect to set-inclusion relation of sets of attributes. Intuitively, adding new attributes will improve the definability, that is, we have more definable sets. It follows that different granularities of P-definable sets can be established and P-definable sets may be used in granular computing [37].

Similarly, we can obtain three categories of P-definable sets and all the properties we discussed for AT-definable sets hold for P-definable sets.

Suppose $P = \{a_1, a_2, \cdots, a_k\} \subseteq AT$. A P-definable basic set can be represented as:

$$X = m(a = v), \tag{2.25}$$

where $a \in P$ and $v \in V_a$.

Consider a special form of formulas represented as follows:

$$(a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_k = v_k), \tag{2.26}$$

where $a_i \in P$, $v_i \in V_{a_i}$, and $a_i \neq a_j$ when $i \neq j$ $(1 \leq i, j \leq k)$. That is, the formula takes exactly one atomic formula from each attribute in P. Such formulas are called P-minterms. A subset $X \subseteq U$ is called a P-definable elementary set if there exists a P-minterm t_P such that $X = m(t_P)$.

A P-conjunctively definable set is defined by a formula containing only logic conjunction and attributes from P. Such a set $X \subseteq U$ can be represented as:

$$X = m\big((a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_r = v_r)\big), \tag{2.27}$$

where $a_i \in P$, and $v_i \in V_{a_i}$ $(1 \leq i \leq r)$. The family $\mathrm{CDEF}_P(U)$ denotes all P-conjunctively definable sets. By definitions, we know that $\mathrm{CDEF}(U)$=$\mathrm{CDEF}_{AT}(U)$.

The first property is that P-definable basic sets and P-definable elementary sets are special cases of P-conjunctively definable sets. The second one is that a P-conjunctively definable set can be interpreted as the intersection of a family of P-definable basic sets. The third one is that a P-conjunctively definable set can be interpreted as the union of a family of P-definable elementary sets. The fourth one is that the set of all P-definable sets, that is, $\mathrm{DEF}_P(U)$, is an atomic Boolean algebra.

2.4 Rough Set Approximations

In rough set theory, an undefinable set cannot be defined by a formula in DL or a family of sets of attribute-value pairs. In order to get a sense about the meaning of an undefinable set, definable sets are used to approximate it and rough set approximations are introduced. In this section, the original definition of rough set approximations proposed by Pawlak [24, 26] is reviewed.

For a given set $X \subseteq U$, we can find several subsets and supersets of X which are definable. Let $\mathbb{Y} = \{\emptyset, Y_1, Y_2, \cdots, Y_m\}$ denote all the definable subsets of X and $\mathbb{Y}' = \{Y_1', Y_2', \cdots, Y_n', U\}$ denote all the definable supersets of X. Intuitively, all these definable sets can be used to approximately interpret the set X. In order to get the most precise description of X, the set in $\mathbb{Y}$ with the maximum number of objects and the set in $\mathbb{Y}'$ with the minimum number of objects will be the best choices.

Definition 12. *Suppose $X \in U$ is the set to be approximated. Its lower approximation $\underline{apr}(X)$ and upper approximation $\overline{apr}(X)$ [24, 26] are defined as:*

$$\begin{aligned}
\underline{apr}(X) &= \textit{the greatest (largest) definable set that is contained in } X; \\
\overline{apr}(X) &= \textit{the least (smallest) definable set that contains } X.
\end{aligned}$$

By definition, both approximations are unique. One can obtain the following properties of approximations [24, 26].

$$\begin{array}{ll}
\text{(P1)} & \underline{apr}(X) \subseteq X \subseteq \overline{apr}(X), \\
\text{(P2)} & \underline{apr}(X) = X = \overline{apr}(X) \iff X \text{ is definable}, \\
\text{(P3)} & \underline{apr}(\emptyset) = \emptyset = \overline{apr}(\emptyset), \\
 & \underline{apr}(U) = U = \overline{apr}(U), \\
\text{(P4)} & \underline{apr}(\underline{apr}(X)) = \underline{apr}(X) = \overline{apr}(\underline{apr}(X)), \\
 & \underline{apr}(\overline{apr}(X)) = \overline{apr}(X) = \overline{apr}(\overline{apr}(X)), \\
\text{(P5)} & A \subseteq B \implies \underline{apr}(A) \subseteq \underline{apr}(B), \\
 & A \subseteq B \implies \overline{apr}(A) \subseteq \overline{apr}(B), \\
\text{(P6)} & \underline{apr}(A \cap B) = \underline{apr}(A) \cap \underline{apr}(B), \\
 & \overline{apr}(A \cup B) = \overline{apr}(A) \cup \overline{apr}(B), \\
\text{(P7)} & \underline{apr}(A \cup B) \supseteq \underline{apr}(A) \cup \underline{apr}(B), \\
 & \overline{apr}(A \cap B) \subseteq \overline{apr}(A) \cap \overline{apr}(B), \\
\text{(P8)} & \underline{apr}(X) = (\overline{apr}(X^c))^c, \\
 & \overline{apr}(X) = (\underline{apr}(X^c))^c.
\end{array}$$

Properties from (P1) to (P4) can be easily verified by using the definition of approximations. The other four properties show a duality between lower and upper approximations [41].

By now, a general framework of concept analysis with rough set theory can be established as given by Figure 2.6. Generally, classification rules are given by components in the approximations, and the components may have different forms due to different formulations of the approximations. A lower approximation gives certain rules, while an upper approximation gives uncertain rules which may lead to misclassifications. We will discuss how to create classification rules from different formulations of approximations in Section 4.6.

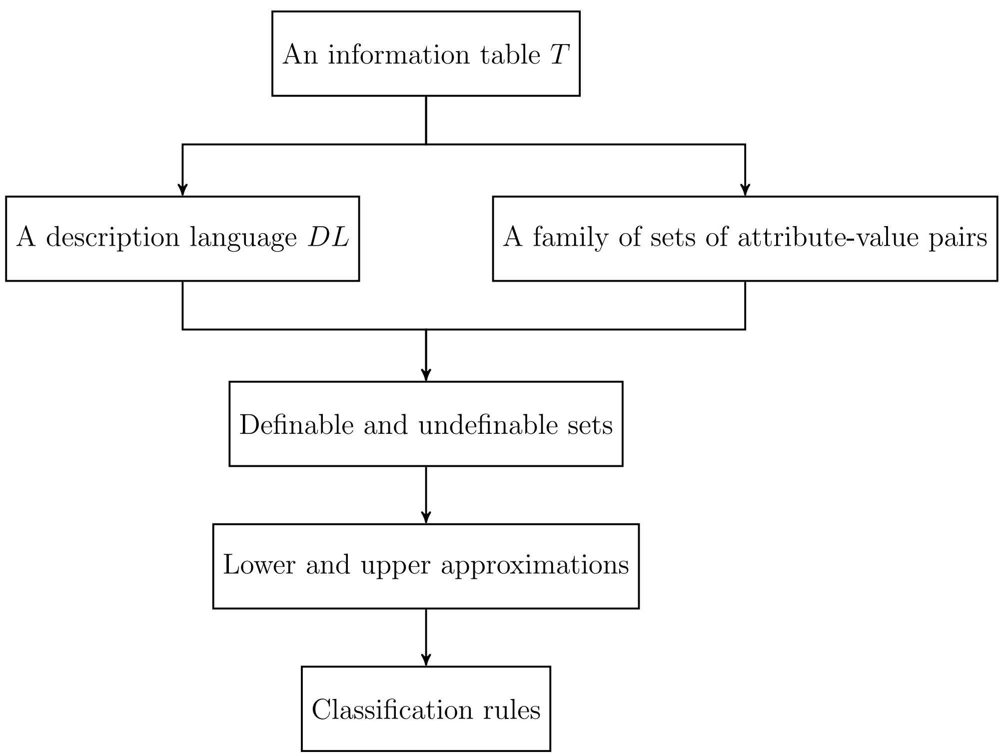


Figure 2.6: A General Framework of Concept Analysis with Rough Set Theory

A main advantage of this framework is that it provides a semantically sound model by answering why we need approximations and how to define the approximations. The disadvantage of this framework is that it is difficult to construct and compute the family of all definable sets and the approximations. Therefore, this formulation is semantically sound but computationally weak.

The following chapters discuss several computational models and formulations of approximations from the viewpoint of structural information.

2.5 Summary

This chapter reviews basic notions in concept analysis with rough set theory, including the notions of concepts, concept representation in an information table, definability of sets and approximations.

According to the classical view, a concept is composed of two parts: the intension and extension. Based on an information table in rough set theory, intension may be formally represented by a formula in a description language DL or equivalently by a family of sets of attribute-value pairs, and extension can be represented by a subset of objects.

Due to limited attributes and information in an information table, one can obtain two kinds of sets of objects: definable and undefinable sets. Intuitively, a definable set can be precisely defined or described by a formula or a family of sets of attribute-value pairs, while an undefinable set cannot be. A concept is definable if its extension is a definable set, while it is undefinable if its extension is undefinable. The undefinability leads to an important construct in rough set theory, that is, the lower and upper approximations. This chapter reviews the definition of rough set approximations, which mainly focuses on semantics of approximations rather than computation. The next chapter will discuss two computational formulations of rough set approximations and analyze their structural information.

Moreover, this chapter discusses three categories of definable sets and investigate their properties. It also generalizes the notions of definability with respect to a subset of attributes $P \subseteq AT$.

Chapter 3

TWO TYPES OF ROUGH SET APPROXIMATIONS CONSIDERING STRUCTURAL INFORMATION

Two types of formulations of rough set approximations containing structural information are examined in this chapter. They are Pawlak approximations and adaptive approximations. We present an analysis about the advantages and disadvantages of these two types of approximations in terms of their structural information.

3.1 Unstructured and Structured Pawlak Approximations

Pawlak [24,26] proposed the formulation of the two approximations using equivalence classes induced by an equivalence relation. This section reviews Pawlak approximations and discusses its structural information.

3.1.1 Equivalence Relations and Equivalence Classes

Consider a finite and nonempty universe U. An equivalence relation on U is a binary relation $E \subseteq U \times U$ satisfying three properties:

(1) E is reflexive:

$$\forall\, x \in U,\ xEx.$$

(2) E is symmetric:

$$\forall\, x, y \in U,\ xEy \implies yEx.$$

(3) E is transitive:

$$\forall\, x, y, z \in U,\ (xEy) \wedge (yEz) \implies xEz.$$

In an information table, an equivalence relation is defined with respect to a subset of attributes [24, 26].

Definition 13. *Given a subset of attributes $A \subseteq AT$ in an information table, an equivalence relation is defined as:*

$$E_A = \{(x, y) \in U \times U \mid \forall\, a \in A, I_a(x) = I_a(y)\}. \tag{3.1}$$

In the case that a single attribute a is considered, we use E_a and $E_{\{a\}}$ interchangeably. The equivalence relation E_A can be constructed by the intersection of equivalence relations defined by single attributes in A. That is:

$$E_A = \bigcap_{a \in A} E_a. \tag{3.2}$$

Given a subset of attributes, an equivalence relation can be constructed. Let $|AT|$ denote the cardinality of AT, that is, the number of all attributes. There are totally $2^{|AT|}$ subsets of attributes and each of them can induce an equivalence relation. By noticing that different subsets of attributes may induce the same equivalence relation, there are at most a total of $2^{|AT|}$ equivalence relations, which may form a lattice. Suppose the set of attributes is $AT = \{a, b, c\}$. If equivalence relations defined by different subsets of attributes are different, then the lattice is given by Figure 3.1. The equivalence relations in higher levels can be formed by taking set intersection of the equivalence relations in lower levels.

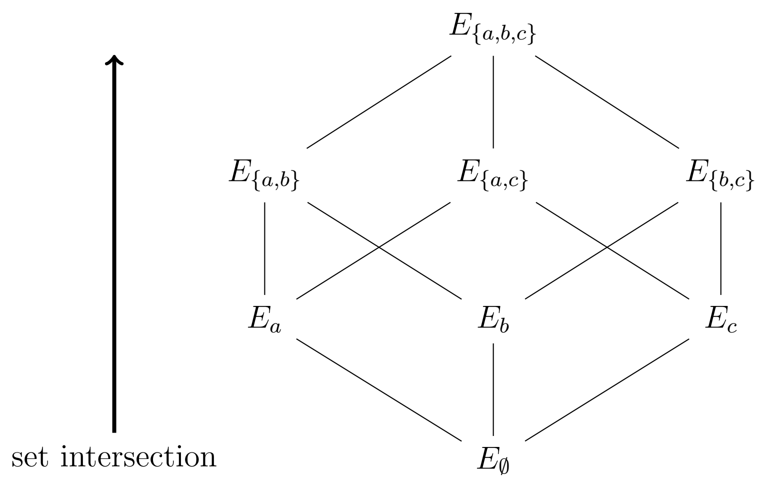


Figure 3.1: A Lattice Formed by Equivalence Relations

For an object $x \in U$, we can form its equivalence class $[x]_{E_A}$ as the set of all objects equivalent to x, that is:

$$[x]_{E_A} = \{y \in U \mid xE_Ay\}. \tag{3.3}$$

Objects in one equivalence class induce the same equivalence class, that is:

$$\forall\, y \in [x]_{E_A},\ [y]_{E_A} = [x]_{E_A}. \tag{3.4}$$

All equivalence classes form a partition of the universe U. Generally, a partition is defined as a family of nonempty and pairwise disjoint sets with their union being the universe. Each set in a partition is called a block. Suppose $\Pi \subseteq 2^U$ denotes a partition of a set U. The following two properties are satisfied:

$$
\begin{aligned}
&(1) \qquad \forall X, Y \in \Pi,\ (X = Y) \vee (X \cap Y = \emptyset),\\
&(2) \qquad \bigcup_{X \in \Pi} X = U.
\end{aligned}
\tag{3.5}
$$

There is a one-to-one correspondence between the set of all equivalence relations and the set of all partitions of U. The partition formed by the equivalence classes can be viewed as a quotient set of U. We denote the partition as:

$$
U/E_A = \{[x]_{E_A} \mid x \in U\}. \tag{3.6}
$$

An equivalence class is a conjunctively definable set. Given an equivalence class $[x]_{E_A}$, we have:

$$
[x]_{E_A} \quad = \quad m\big(\bigwedge_{a \in A} a = I_a(x)\big). \tag{3.7}
$$

By considering all attributes, the equivalence classes induced are definable elementary sets:

$$
[x]_{E_{AT}} \quad = \quad m\big(\bigwedge_{a \in AT} a = I_a(x)\big). \tag{3.8}
$$

Set inclusion relationship establishes a partial order on the set of all equivalence relations. Suppose we have two sets of attributes $A, B \subseteq AT$. Their induced equivalence relations E_A and E_B satisfy the following property:

$$
A \subseteq B \implies E_A \supseteq E_B. \tag{3.9}
$$

Consequently, we can define a partial order $\preccurlyeq$ on the set of all partitions of the universe $(\{U/E_A \mid A \subseteq AT\}, \preccurlyeq)$ with respect to an information table.

Definition 14. *Given two partitions U/E_A and U/E_B, the partial order $\preccurlyeq$ is defined* *as:*

$$U/E_B \preccurlyeq U/E_A \iff \forall\,[y] \in U/E_B,\ \exists\,[x] \in U/E_A,\ \textit{such that}\ [y] \subseteq [x]. \quad (3.10)$$

By definition, it can be concluded that:

$$A \subseteq B \implies E_B \subseteq E_A \iff U/E_B \preccurlyeq U/E_A. \quad (3.11)$$

E_B is called a finer equivalence relation and U/E_B a finer partition. Equivalence classes in U/E_B are considered finer in the sense that they are subsets of objects in equivalence classes of U/E_A .

Two different sets of attributes may induce the same equivalence relation. Therefore, an equivalence relation $\equiv$ can be defined on the family of all subsets of attributes AT.

Definition 15. *Suppose $A, B \subseteq AT$. An equivalence relation $\equiv$ is defined as:*

$$A \equiv B \iff E_A = E_B. \quad (3.12)$$

That is, A and B are equivalent if they induce the same equivalence relation.

There may be a subset of attributes which induces the same equivalence relation with all attributes AT. If the subset is minimal, it is called an attribute reduct [24, 49–51, 53].

3.1.2 Definition of Unstructured and Structured Pawlak Approximations

Pawlak [24, 26] defined the formulation of approximations as unions of equivalence classes. Suppose $X \subseteq U$ is a set to be approximated and E_A is an equivalence relation.

The Pawlak approximations are formulated as:

$$\begin{aligned} \underline{apr}_{E_A}(X) &= \bigcup\{[x]_{E_A} \in U/E_A \mid [x]_{E_A} \subseteq X\}, \\ \overline{apr}_{E_A}(X) &= \bigcup\{[x]_{E_A} \in U/E_A \mid [x]_{E_A} \cap X \neq \emptyset\}. \end{aligned} \tag{3.13}$$

Bryniarski [3] proposed to remove the union operator in Pawlak's formulation and define the approximations as:

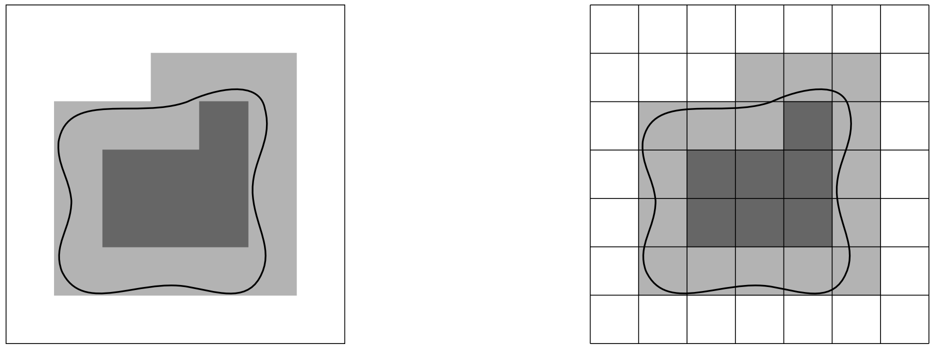
$$\begin{aligned} \underline{bapr}_{E_A}(X) &= \{[x]_{E_A} \in U/E_A \mid [x]_{E_A} \subseteq X\}, \\ \overline{bapr}_{E_A}(X) &= \{[x]_{E_A} \in U/E_A \mid [x]_{E_A} \cap X \neq \emptyset\}. \end{aligned} \tag{3.14}$$

By retaining the individual equivalence classes, Bryniarski's definition keeps structural information which was missing in Pawlak's definition. In the following discussion, we may use $\underline{apr}(X)$, $\overline{apr}(X)$, $\underline{bapr}(X)$, and $\overline{bapr}(X)$ instead of $\underline{apr}_{E_A}(X)$, $\overline{apr}_{E_A}(X)$, $\underline{bapr}_{E_A}(X)$, and $\overline{bapr}_{E_A}(X)$, respectively, if the discussion does not depend on specific equivalence relations. We call $(\underline{apr}(X), \overline{apr}(X))$ the unstructured Pawlak approximations, and $(\underline{bapr}(X), \overline{bapr}(X))$ Bryniarski approximations or the structured Pawlak approximations with each equivalence class being considered as a granule. This point of view connects rough set theory and granular computing [37].

There may exist subsets of AT which are minimal sets of attributes that produces the same partition of U and the same Pawlak approximations with all attributes AT. Such subsets are called attribute reducts [24, 49–51, 53].

Figure 3.2 gives an illustration of both approximations with each small square representing an equivalence class. Squares in the same size are used to show that they are induced by the same equivalence relation. They do not necessarily contain the same number of objects.

By comparing Figures 3.2 (a) and 3.2 (b), one can easily observe the advantage of structural information retained by preserving individual equivalence classes. The



(a) Unstructured Pawlak Approximations (b) Structured Pawlak Approximations

A set X of objects lower approximation upper approximation - lower approximation

Figure 3.2: Unstructured and Structured Pawlak Approximations

unstructured Pawlak approximations do not explicitly show which equivalence classes are used. They are, however, explicitly shown in the structured Pawlak approximations.

By definitions, $\underline{apr}(X)$ and $\overline{apr}(X)$ are subsets of U, while $\underline{bapr}(X)$ and $\overline{bapr}(X)$ are subsets of the partition U/E_A. Although these two formulations are slightly different in form, they can define each other as:

$$\begin{aligned} \underline{apr}(X) &= \bigcup \underline{bapr}(X), \\ \overline{apr}(X) &= \bigcup \overline{bapr}(X); \end{aligned} \tag{3.15}$$

and

$$\begin{aligned} \underline{bapr}(X) &= \{[x]_{E_A} \in U/E_A \mid [x]_{E_A} \subseteq \underline{apr}(X)\}, \\ \overline{bapr}(X) &= \{[x]_{E_A} \in U/E_A \mid [x]_{E_A} \subseteq \overline{apr}(X)\}. \end{aligned} \tag{3.16}$$

Based on equivalence relations and equivalence classes, Pawlak proposed a computational model. As a comparison with the general semantically sound model given by Figure 2.6, Figure 3.3 gives a framework of Pawlak's computational model, which is equivalent to the semantically sound model. The family of all definable sets is equivalent to the family of all equivalence classes and all possible union sets [42]. The approximations defined in Definition 12 are equivalent to Pawlak approximations in the sense that they cover the same sets of objects.

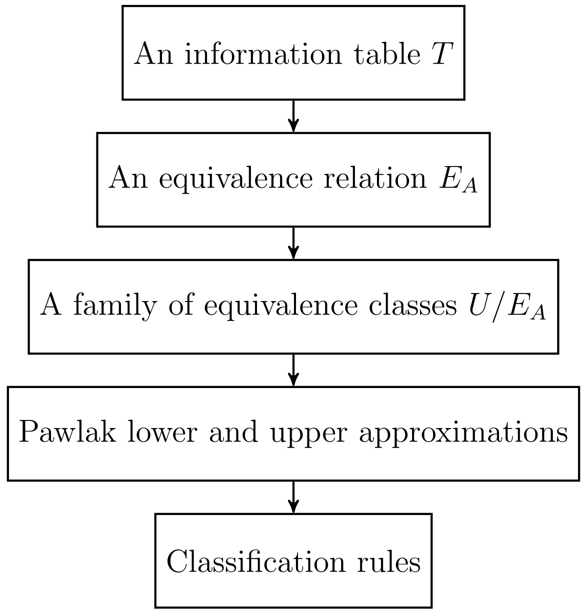


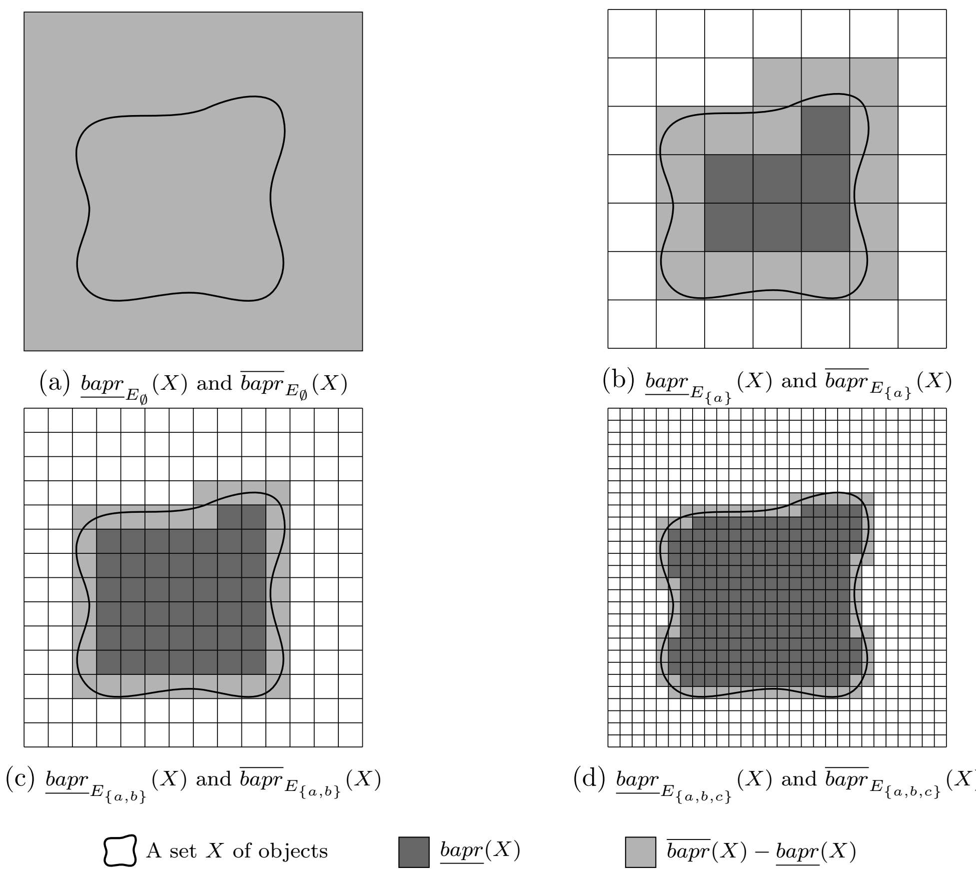
Figure 3.3: Pawlak's Computational Model

3.1.3 Structural Information in Structured Pawlak Approximations

By removing the union operator, the structured Pawlak approximations explicitly give the equivalence classes used, which is called the structural information since the equivalence classes can express the internal structure of approximations.

Although structural information is provided in the structured Pawlak approximations, it may not be optimal. By using different sets of attributes, we can construct different granularities of the structured Pawlak approximations. Suppose the set of attributes is $AT = \{a, b, c\}$. Figure 3.4 shows the four possible granularities with each caption indicating the corresponding equivalence relation used.

As shown by Figure 3.4, by using more attributes, the equivalence classes used to construct the approximations become finer. As a consequence, the approximations constructed become more precise. However, finer equivalence classes are not preferred in practice, as they require the use of more attributes and create less general classification rules. In a structured Pawlak approximation, it may happen that some finer equivalence classes can be combined into a coarser equivalence class that uses fewer attributes and gives a more general classification rule. This may be a disadvantage of the structured Pawlak approximation.



(a) $\underline{bapr}_{E_\emptyset}(X)$ and $\overline{bapr}_{E_\emptyset}(X)$

(b) $\underline{bapr}_{E_{\{a\}}}(X)$ and $\overline{bapr}_{E_{\{a\}}}(X)$

(c) $\underline{bapr}_{E_{\{a,b\}}}(X)$ and $\overline{bapr}_{E_{\{a,b\}}}(X)$

(d) $\underline{bapr}_{E_{\{a,b,c\}}}(X)$ and $\overline{bapr}_{E_{\{a,b,c\}}}(X)$

A set X of objects $\underline{bapr}(X)$ $\overline{bapr}(X) - \underline{bapr}(X)$

Figure 3.4: Structured Pawlak Approximations with Different Granularities

3.1.4 An Example of Pawlak Approximations

Consider Table 2.1 and a set $X = \{o_1, o_2, o_3, o_4, o_5\}$. Three granularities of unstructured and structured Pawlak approximations are constructed by considering three subsets of attributes {Height}, {Height,Hair}, and {Height, Hair, Eyes}$= AT$, respectively. The three partitions are:

$$
\begin{aligned}
U/E_{\{\text{Height}\}} &= \big\{\{o_1, o_2, o_3\}, \{o_4, o_5, o_6\}\big\},\\
U/E_{\{\text{Height,Hair}\}} &= \big\{\{o_1, o_2\}, \{o_3\}, \{o_4\}, \{o_5, o_6\}\big\},\\
U/E_{AT} &= \big\{\{o_1\}, \{o_2\}, \{o_3\}, \{o_4\}, \{o_5, o_6\}\big\}.
\end{aligned}
$$

Pawlak approximations with different granularities are constructed as:

(1) Pawlak approximations based on $E_{\{\text{Height}\}}$:

$$
\begin{aligned}
\underline{apr}_1(X) &= \bigcup\{K \in U/E_{\{\text{Height}\}} \mid K \subseteq X\}\\
&= \{o_1, o_2, o_3\},\\
\overline{apr}_1(X) &= \bigcup\{K \in U/E_{\{\text{Height}\}} \mid K \cap X \neq \emptyset\}\\
&= \{o_1, o_2, o_3, o_4, o_5, o_6\};
\end{aligned}
$$

and

$$
\begin{aligned}
\underline{bapr}_1(X) &= \{K \in U/E_{\{\text{Height}\}} \mid K \subseteq X\}\\
&= \big\{\{o_1, o_2, o_3\}\big\},\\
\overline{bapr}_1(X) &= \{K \in U/E_{\{\text{Height}\}} \mid K \cap X \neq \emptyset\}\\
&= \big\{\{o_1, o_2, o_3\}, \{o_4, o_5, o_6\}\big\}.
\end{aligned}
$$

(2) Pawlak approximations based on $E_{\{\text{Height,Hair}\}}$:

$$
\begin{aligned}
\underline{apr}_2(X) &= \bigcup\{K \in U/E_{\{\text{Height,Hair}\}} \mid K \subseteq X\} \\
&= \{o_1, o_2, o_3, o_4\}, \\
\overline{apr}_2(X) &= \bigcup\{K \in U/E_{\{\text{Height,Hair}\}} \mid K \cap X \neq \emptyset\} \\
&= \{o_1, o_2, o_3, o_4, o_5, o_6\};
\end{aligned}
$$

and

$$
\begin{aligned}
\underline{bapr}_2(X) &= \{K \in U/E_{\{\text{Height,Hair}\}} \mid K \subseteq X\} \\
&= \big\{\{o_1, o_2\}, \{o_3\}, \{o_4\}\big\}, \\
\overline{bapr}_2(X) &= \{K \in U/E_{\{\text{Height,Hair}\}} \mid K \cap X \neq \emptyset\} \\
&= \big\{\{o_1, o_2\}, \{o_3\}, \{o_4\}, \{o_5, o_6\}\big\}.
\end{aligned}
$$

(3) Pawlak approximations based on E_{AT}:

$$
\begin{aligned}
\underline{apr}_3(X) &= \bigcup\{K \in U/E_{AT} \mid K \subseteq X\} \\
&= \{o_1, o_2, o_3, o_4\}, \\
\overline{apr}_3(X) &= \bigcup\{K \in U/E_{AT} \mid K \cap X \neq \emptyset\} \\
&= \{o_1, o_2, o_3, o_4, o_5, o_6\};
\end{aligned}
$$

and

$$
\begin{aligned}
\underline{bapr}_3(X) &= \{K \in U/E_{AT} \mid K \subseteq X\} \\
&= \big\{\{o_1\}, \{o_2\}, \{o_3\}, \{o_4\}\big\}, \\
\overline{bapr}_3(X) &= \{K \in U/E_{AT} \mid K \cap X \neq \emptyset\} \\
&= \big\{\{o_1\}, \{o_2\}, \{o_3\}, \{o_4\}, \{o_5, o_6\}\big\}.
\end{aligned}
$$

3.2 Adaptive Approximations

Deng, Wang and Yao [6] proposed the formulation of the two approximations by using a sequence of equivalence relations, called the adaptive approximations. Since this formulation contains structural information, we give a review of its ideas and analyze its structural information.

3.2.1 An Approach to Construct Adaptive Approximations

In the structured Pawlak approximations, only one granularity of equivalence relations is used. The adaptive approximations consider a sequence of subsets of attributes which induce different granularities of equivalence relations. These equivalence relations are used one by one to construct the adaptive approximations.

A sequence of subsets of attributes is defined as $Seq_n = \{A_1, A_2, ..., A_n\}$, where $A_i \subseteq AT$ $(1 \leq i \leq n)$ and all the sets in the sequence have the following relationship:

$$A_1 \subsetneq A_2 \subsetneq \cdots \subsetneq A_n. \tag{3.17}$$

Consequently, we can get a sequence of equivalence relations satisfying the condition:

$$E_{A_n} \subseteq E_{A_{n-1}} \subseteq \cdots \subseteq E_{A_2} \subseteq E_{A_1}. \tag{3.18}$$

The corresponding partitions in different granularities satisfy the condition:

$$U/E_{A_n} \preccurlyeq U/E_{A_{n-1}} \preccurlyeq \cdots \preccurlyeq U/E_{A_2} \preccurlyeq U/E_{A_1}. \tag{3.19}$$

Suppose X is the set to be approximated. The adaptive approximations can be constructed recursively as follows:

(1) Initialization: Use the partition $U/E_{A_0} = U/E_{\emptyset} = \{U\}$ to compute the structured

Pawlak approximates of X:

$$\begin{aligned}\underline{dapr}_0(X) &= \{[x]_{E_{A_0}} \mid [x]_{E_{A_0}} \in U/E_{A_0},\ [x]_{E_{A_0}} \subseteq X\} \\ &= \emptyset, \\ \overline{dapr}_0(X) &= \{[x]_{E_{A_0}} \mid [x]_{E_{A_0}} \in U/E_{A_0},\ [x]_{E_{A_0}} \cap X \neq \emptyset\} \\ &= \{U\}. \end{aligned} \tag{3.20}$$

The difference set given by objects in these two approximations includes objects to be partitioned in the next step:

$$\begin{aligned} X_1 &= \bigcup \overline{dapr}_0(X) - \bigcup \underline{dapr}_0(X) \\ &= U. \end{aligned} \tag{3.21}$$

(2) Recursion: Let $(\underline{dapr}_{i-1}(X), \overline{dapr}_{i-1}(X))$ denote the approximations constructed in the $(i-1)$th step. In the ith step, equivalence relation E_{A_i} is used to partition $X_i = \bigcup \overline{dapr}_{i-1}(X) - \bigcup \underline{dapr}_{i-1}(X)$. This partition is denoted as X_i/E_{A_i}. The adaptive approximations in the ith step are constructed as:

$$\begin{aligned} \underline{dapr}_i(X) &= \underline{dapr}_{i-1}(X) \cup \{K \in X_i/E_{A_i} \mid K \subseteq X\}, \\ \overline{dapr}_i(X) &= \underline{dapr}_i(X) \cup \{K \in X_i/E_{A_i} \mid K \cap X \neq \emptyset\}. \end{aligned} \tag{3.22}$$

Objects to be partitioned in the next step are:

$$X_{i+1} = \bigcup \overline{dapr}_i(X) - \bigcup \underline{dapr}_i(X). \tag{3.23}$$

According to the construction of the adaptive approximations, one can conclude that:

$$\underline{dapr}_i(X) \subseteq X \subseteq \overline{dapr}_i(X) \quad (1 \leq i \leq n). \tag{3.24}$$

Suppose the set of attributes is $AT = \{a, b, c\}$ and the sequence of subsets of attributes is $Seq_3 = \{\{a\}, \{a, b\}, \{a, b, c\}\}$. Figure 3.5 illustrates the approach to construct the adaptive approximations. The captions indicate the equivalence relations used in the corresponding steps. Squares with the same size represent equivalence classes induced from the same equivalence relation. With more attributes to construct the equivalence relation, the equivalence classes become finer and smaller squares are used.

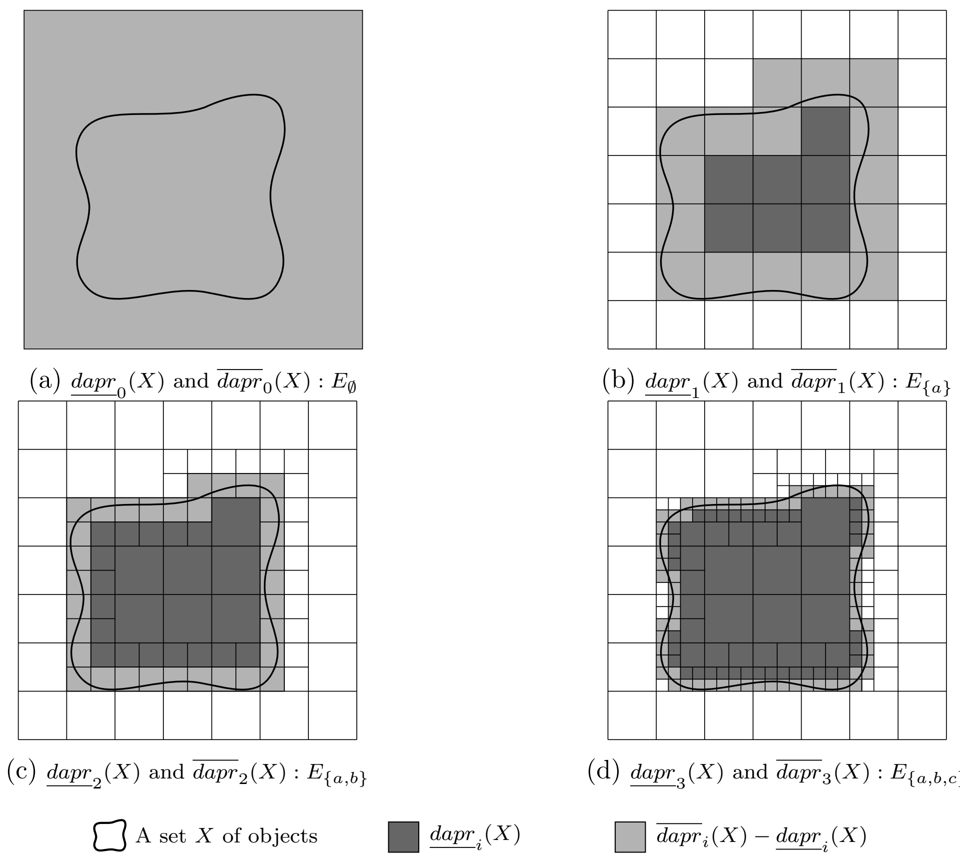


Figure 3.5: An Illustration of Constructing Adaptive Approximations

3.2.2 Relationships to Pawlak Approximations

Suppose the sequence $Seq_n = \{A_1, A_2, \cdots, A_n\}$ is used to construct the adaptive approximations. The adaptive approximations cover the same sets of objects as the Pawlak approximations induced by equivalence relation E_{A_n}.

Theorem 5. *Suppose the adaptive approximations are constructed by using* $Seq_n = \{A_1, A_2, \cdots, A_n\}$. *The following properties hold:*

$$
\begin{array}{lrclcl}
(1) & \bigcup \underline{dapr}_n(X) & = & \bigcup \underline{bapr}_{E_{A_n}}(X) & = & \underline{apr}_{E_{A_n}}(X), \\
(2) & \bigcup \overline{dapr}_n(X) & = & \bigcup \overline{bapr}_{E_{A_n}}(X) & = & \overline{apr}_{E_{A_n}}(X).
\end{array} \tag{3.25}
$$

This theorem can be verified by the definition of Pawlak approximations and the construction of adaptive approximations. It is intuitively shown by Figure 3.6. The proof is given in Section A.5 in Appendix A.

Coarser equivalence classes that have a larger number of objects are preferred in the adaptive approximations, while the structured Pawlak approximations only use the finest ones induced by E_{A_n}. Based on the relationship between different granularities of partitions:

$$
U/E_{A_n} \preccurlyeq U/E_{A_{n-1}} \preccurlyeq \cdots \preccurlyeq U/E_{A_1} \preccurlyeq U/E_{A_0}, \tag{3.26}
$$

it follows that:

$$
\begin{array}{ll}
(1) & \forall\, K \in \underline{bapr}_{E_{A_n}}(X),\ \exists\, K' \in \underline{dapr}(X),\ \text{such that } K \subseteq K', \\
(2) & \forall\, K \in \overline{bapr}_{E_{A_n}}(X),\ \exists\, K' \in \overline{dapr}(X),\ \text{such that } K \subseteq K'.
\end{array} \tag{3.27}
$$

Figure 3.6 explicitly gives the relationships between different granularities of structured Pawlak approximations and the adaptive approximations by combining Figure 3.4 and Figure 3.5 and omitting the approximations induced by $E_\emptyset$. It shows that the structured Pawlak approximations and the corresponding adaptive approximations cover the same sets of objects.

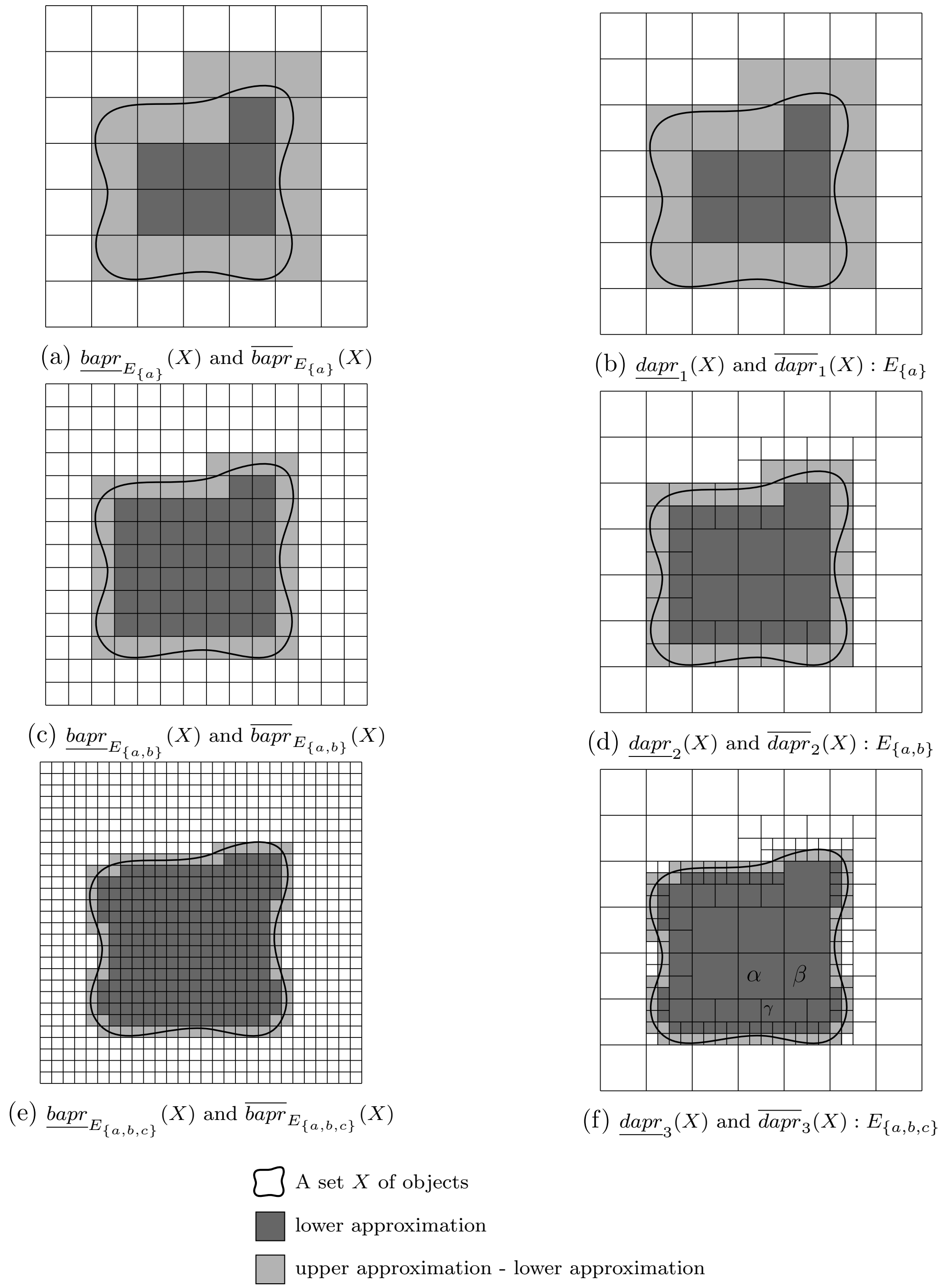


Figure 3.6: Structured Pawlak Approximations and Adaptive Approximations with Different Granularities

3.2.3 Structural Information in Adaptive Approximations

Although the adaptive approximations have a better structure than Pawlak approximations as a result of preferring coarser equivalence classes, they have some disadvantages nonetheless. One possible problem is that several equivalence classes with different granularities can be combined into a larger definable set. For example, in Figure 3.6 (f), the union of equivalence classes α and γ may be a definable set, and the union of β and γ may also be a definable set. These two definable sets have more objects and may create more general rules which are preferred in concept analysis. The main reason is that the adaptive approximations are confined by the partition-based framework which does not allow overlap between the definable sets in an approximation.

3.2.4 An Example of Adaptive Approximations

The same example as in Section 3.1.4 is used. Consider Table 2.1 and set $X = \{o_1, o_2, o_3, o_4, o_5\}$. The sequence of attribute sets is $Seq_4 = \big\{$ {Height}, {Height, Hair}, {Height, Hair, Eyes}$\big\}$.

(1) For the initialization, equivalence relation $E_\emptyset$ is used and the approximations are:

$$\begin{aligned} \underline{dapr}_0(X) &= \emptyset, \\ \overline{dapr}_0(X) &= \big\{U\big\} \\ &= \big\{\{o_1, o_2, o_3, o_4, o_5, o_6\}\big\}. \end{aligned}$$

It follows that:

$$X_1 = \bigcup \overline{dapr}_0(X) - \bigcup \underline{dapr}_0(X) = \{o_1, o_2, o_3, o_4, o_5, o_6\}.$$

(2) In the first step, {Height} is used and the partition is:

$$X_1/E_{\{\text{Height}\}} = \big\{\{o_1, o_2, o_3\}, \{o_4, o_5, o_6\}\big\}.$$

Therefore, the approximations are:

$$\begin{aligned}
\underline{dapr}_1(X) &= \underline{dapr}_0(X) \cup \{K \in X_1/E_{\{\text{Height}\}} \mid K \subseteq X\} \\
&= \big\{\{o_1, o_2, o_3\}\big\}, \\
\overline{dapr}_1(X) &= \underline{dapr}_1(X) \cup \{K \in X_1/E_{\{\text{Height}\}} \mid K \cap X \neq \emptyset\} \\
&= \big\{\{o_1, o_2, o_3\}, \{o_4, o_5, o_6\}\big\}.
\end{aligned}$$

The set X_2 is computed as:

$$X_2 \;=\; \bigcup \overline{dapr}_1(X) - \bigcup \underline{dapr}_1(X) \;=\; \{o_4, o_5, o_6\}.$$

(3) In the second step, $\{\text{Height}, \text{Hair}\}$ is used and the partition is:

$$X_2/E_{\{\text{Height},\text{Hair}\}} = \big\{\{o_4\}, \{o_5, o_6\}\big\}.$$

The approximations are:

$$\begin{aligned}
\underline{dapr}_2(X) &= \underline{dapr}_1(X) \cup \{K \in X_2/E_{\{\text{Height},\text{Hair}\}} \mid K \subseteq X\} \\
&= \big\{\{o_1, o_2, o_3\}, \{o_4\}\big\}, \\
\overline{dapr}_2(X) &= \underline{dapr}_2(X) \cup \{K \in X_2/E_{\{\text{Height},\text{Hair}\}} \mid K \cap X \neq \emptyset\} \\
&= \big\{\{o_1, o_2, o_3\}, \{o_4\}, \{o_5, o_6\}\big\}.
\end{aligned}$$

It follows that:

$$X_3 \;=\; \bigcup \overline{dapr}_2(X) - \bigcup \underline{dapr}_2(X) \;=\; \{o_5, o_6\}.$$

(4) In the third step, {Height, Hair, Gender} is used and the partition is:

$$X_3/E_{\{\text{Height,Hair,Eyes}\}} = \big\{\{o_5, o_6\}\big\}.$$

The approximations are:

$$\begin{aligned}
\underline{dapr}_3(X) &= \underline{dapr}_2(X) \cup \{K \in X_3/E_{\{\text{Height,Hair,Eyes}\}} \mid K \subseteq X\} \\
&= \big\{\{o_1, o_2, o_3\}, \{o_4\}\big\}, \\
\overline{dapr}_3(X) &= \underline{dapr}_3(X) \cup \{K \in X_3/E_{\{\text{Height,Hair,Eyes}\}} \mid K \cap X \neq \emptyset\} \\
&= \big\{\{o_1, o_2, o_3\}, \{o_4\}, \{o_5, o_6\}\big\}.
\end{aligned}$$

The pair $\big(\underline{dapr}_3(X), \overline{dapr}_3(X)\big)$ is the final adaptive approximations.

3.3 Summary

This chapter gives a further exploration of two formulations of rough set approximations with structural information. While Pawlak approximations use equivalence classes from one equivalence relation, the adaptive approximations use a sequence of equivalence relations from coarser to finer granularities.

On the one hand, Pawlak and adaptive approximations are equivalent in the sense that they cover the same sets of objects. On the other hand, the adaptive approximations may contain better structural information than Pawlak approximations do. By using equivalence relations from coarser to finer ones, the adaptive approximations are able to use the larger equivalence classes first. When learning classification rules from approximations, each equivalence class will give one rule. The larger the equivalence class is, the more general the corresponding rule is.

Although structured Pawlak and adaptive approximations consider structural information, both the approximations are limited by partition-based formulation. That is, they are composed by pairwise disjoint nonempty components. The next chapter will present a new formulation, that is, the structured rough set approximations. It is a covering-based formulation which allows overlap between components.

Chapter 4

STRUCTURED ROUGH SET APPROXIMATIONS

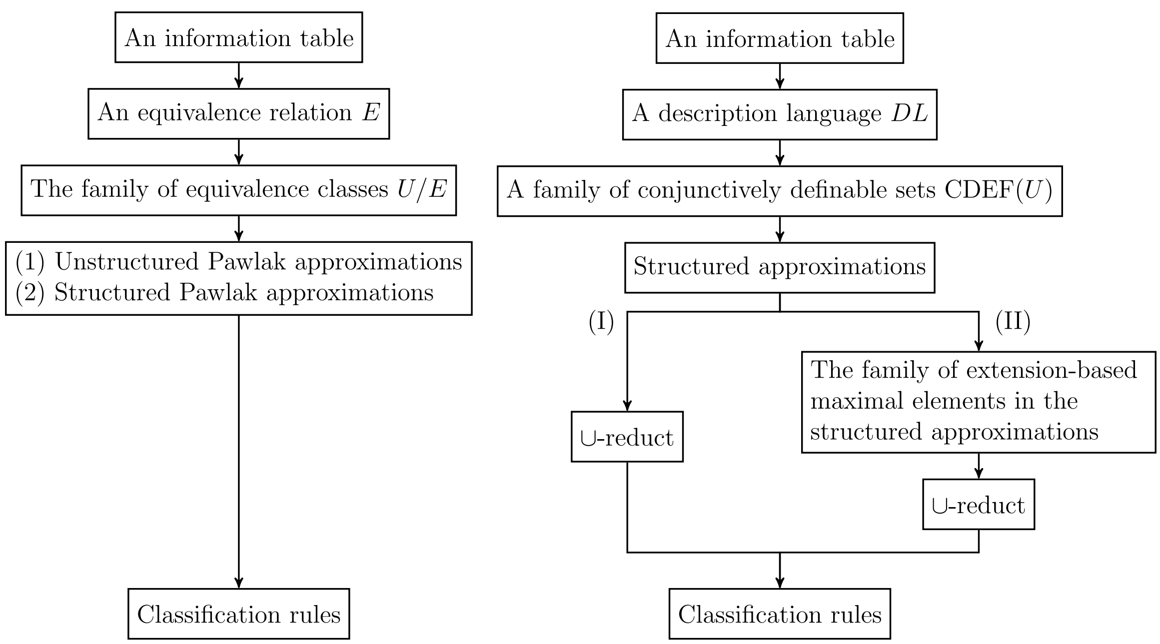
This chapter proposes a new approach to define and construct rough set approximations, called the structured rough set approximations, or structured approximations for short. The notions discussed in this chapter are an extended version of the notions proposed by Yao and Hu [45].

4.1 An Overview of the New Framework

While Pawlak approximations and adaptive approximations are the partition-based approaches, the structured rough set approximations proposed in this study are covering-based. The new approach is proposed in order to obtain a clearer semantical view of approximations and to learn general rules.

Figure 4.1 gives a comparison between the new framework and Pawlak framework. Unlike Pawlak framework, which requires an equivalence relation, the new formulation adopts the description language DL. Semantically, the description language is superior to an equivalence relation since it can explicitly explain the definability of a set. Instead of the family of equivalence classes that are building blocks for Pawlak approximations, the new formulation uses the family of all conjunctively definable sets.

In the Pawlak framework, equivalence classes in an approximation are pairwise disjoint and there do not exist redundant equivalence classes in a structured Pawlak approximation. On the other hand, in the new framework, there may exist redundant conjunctively definable sets in a structured approximation. It is therefore necessary to remove the redundancy. The notion of $\cup$-reduct is adopted from Pawlak [26] and two methods for deriving reduced structured approximations are introduced. The first method, illustrated by branch (I) in Figure 4.1 (b), directly finds a $\cup$-reduct of a structured approximation. The second method, illustrated by branch (II) in Figure 4.1 (b), first computes the family of extension-based maximal elements in a structured approximation and then finds a $\cup$-reduct. One can design other methods to reduce the redundancy as well.



(a) The Pawlak framework (b) The new framework

Figure 4.1: Two Frameworks of Concept Analysis with Rough Set Theory

4.2 A Definition of Structured Rough Set Approximations

In the Pawlak framework, the upper approximation of a set is the complement of the lower approximation of the complement of the set, that is, $\overline{apr}(X) = (\underline{apr}(X^c))^c$. The new framework demands such a duality and only considers the lower approximation.

The family of nonempty sets in $\mathrm{CDEF}(U)$ is a covering of the universe U and, furthermore, $U/E_A \subseteq \mathrm{CDEF}(U)$. By replacing U/E_A in the structured Pawlak lower approximations with $\mathrm{CDEF}(U)$, we propose a generalized definition of structured lower approximations.

Definition 16. *For a subset of objects $X \subseteq U$, its structured lower approximation is defined by:*

$$\underline{sapr}(X) = \{C \in \mathrm{CDEF}(U) \mid C \neq \emptyset,\ C \subseteq X\}. \tag{4.1}$$

That is, $\underline{sapr}(X)$ consists of all nonempty conjunctively definable subsets of X.

The structured lower approximations have a well-defined structure. On the one hand, similarly to the structured Pawlak approximations, we retain the structure expressed by individual conjunctively definable sets. On the other hand, each conjunctively definable set has a clear semantics, which will benefit rule learning in concept analysis with rough set theory.

The structured lower approximations are related to covering-based rough set models [29, 36, 48, 52, 54, 55]. In the Pawlak framework, equivalence classes are definable and formulas can be found to explain the semantics of these equivalence classes. In numerous studies on covering-based rough sets, it is assumed that a covering of the universe is given and various rough set approximations are introduced. However, the semantics of the patches in the covering is not clear. In other words, they use the covering to construct the approximations without explaining why the patches are definable and can be used to approximate other sets. In our formulation, each nonempty set in CDEF(U) is a conjunctively definable set and has a well-defined semantical interpretation expressed by conjunctive formulas. This is consistent with Pawlak formulation in which each equivalence class is a conjunctively definable set. Consequently, our formulation provides a semantically sound approach to covering-based rough set model.

Bonikowski et al. [2] proposed a formulation of rough set approximations which is closely related to our formulation. However, our formulation makes a clearer statement about the structure and semantics of the approximations. Bonikowski et al. [2] defined an approximation space (U, C), where C is a given covering of the universe U. However, they did not explain the semantics of the patches (i.e., subsets) in C. In other words, they used C to approximate a set without answering the question why the patches in C are definable and can be used to construct approximations. In contrast, our formulation considers a specific covering CDEF(U), in which each patch has a clear semantics that can be explained by conjunctive formulas. In addition, they defined the approximations as sets of objects without considering redundancy.

4.3 An Example of Structured Rough Set Approximations

The same example given in Sections 3.1.4 and 3.2.4 is used to demonstrate the basic ideas of structured lower approximations. Consider Table 2.1 and the set $X = \{o_1, o_2, o_3, o_4, o_5\}$.

Firstly, the family of all nonempty conjunctively definable sets with respect to all attributes AT is constructed. They are given by Table 4.1. The conjunctive formulas and the conjunctively definable sets are given separately in two columns.

Conjunctive formulas	**Conjunctively definable sets**
Height=short	$\{o_1, o_2, o_3\}$
Height=tall	$\{o_4, o_5, o_6\}$
Hair=blond	$\{o_1, o_2, o_4\}$
Hair=red	$\{o_3, o_5, o_6\}$
Eyes=blue	$\{o_1, o_3, o_4\}$
Eyes=brown	$\{o_2, o_5, o_6\}$
(Height=short) $\wedge$ (Hair=blond)	$\{o_1, o_2\}$
(Height=short) $\wedge$ (Hair=red) (Hair=red) $\wedge$ (Eyes=blue) (Height=short) $\wedge$ (Hair=red) $\wedge$ (Eyes=blue)	$\{o_3\}$
(Height=tall) $\wedge$ (Hair=blond) (Height=tall) $\wedge$ (Eyes=blue) (Height=tall) $\wedge$ (Hair=blond) $\wedge$ (Eyes=blue)	$\{o_4\}$
(Height=tall) $\wedge$ (Hair=red) (Height=tall) $\wedge$ (Eyes=brown) (Hair=red) $\wedge$ (Eyes=brown) (Height=tall) $\wedge$ (Hair=red) $\wedge$ (Eyes=brown)	$\{o_5, o_6\}$
(Height=short) $\wedge$ (Eyes=blue)	$\{o_1, o_3\}$
(Height=short) $\wedge$ (Eyes=brown) (Hair=blond) $\wedge$ (Eyes=brown) (Height=short) $\wedge$ (Hair=blond) $\wedge$ (Eyes=brown)	$\{o_2\}$
(Hair=blond) $\wedge$ (Eyes=blue)	$\{o_1, o_4\}$
(Height=short) $\wedge$ (Hair=blond) $\wedge$ (Eyes=blue)	$\{o_1\}$

Table 4.1: The Family of All Nonempty Conjunctively Definable Sets

Secondly, according to Definition 16, all nonempty conjunctively definable subsets of X from Table 4.1 are collected to construct $\underline{sapr}(X)$:

$$\begin{aligned}\underline{sapr}(X) \ = \ & \big\{\{o_1, o_2, o_3\}, \{o_1, o_2, o_4\}, \{o_1, o_3, o_4\}, \{o_1, o_2\}, \{o_3\}, \\ & \{o_4\}, \{o_1, o_3\}, \{o_2\}, \{o_1, o_4\}, \{o_1\}\big\}.\end{aligned}$$

Table 4.2 gives the conjunctively definable sets in $\underline{sapr}(X)$ and their corresponding conjunctive formulas.

Conjunctive formulas	**Conjunctively definable sets**
Height=short	$\{o_1, o_2, o_3\}$
Hair=blond	$\{o_1, o_2, o_4\}$
Eyes=blue	$\{o_1, o_3, o_4\}$
(Height=short) ∧ (Hair=blond)	$\{o_1, o_2\}$
(Height=short) ∧ (Hair=red) (Hair=red) ∧ (Eyes=blue) (Height=short) ∧ (Hair=red) ∧ (Eyes=blue)	$\{o_3\}$
(Height=tall) ∧ (Hair=blond) (Height=tall) ∧ (Eyes=blue) (Height=tall) ∧ (Hair=blond) ∧ (Eyes=blue)	$\{o_4\}$
(Height=short) ∧ (Eyes=blue)	$\{o_1, o_3\}$
(Height=short) ∧ (Eyes=brown) (Hair=blond) ∧ (Eyes=brown) (Height=short) ∧ (Hair=blond) ∧ (Eyes=brown)	$\{o_2\}$
(Hair=blond) ∧ (Eyes=blue)	$\{o_1, o_4\}$
(Height=short) ∧ (Hair=blond) ∧ (Eyes=blue)	$\{o_1\}$

Table 4.2: The Structured Lower Approximation of X

4.4 Relationships to Pawlak and Adaptive Approximations

This section investigates the relationships between our structured approximations, Pawlak and adaptive approximations.

Based on the definition of a structured lower approximation and the fact that equivalence classes are conjunctively definable sets, connections can be established between structured lower approximations, unstructured and structured Pawlak lower approximations.

Theorem 6. *The following properties hold:*

$$\begin{aligned} &(1) \qquad \forall\, A \subseteq AT, \;\; \underline{bapr}_{E_A}(X) \subseteq \underline{sapr}(X), \\ &(2) \qquad \underline{apr}_{E_{AT}}(X) = \bigcup \underline{sapr}(X). \end{aligned} \tag{4.2}$$

The proof of this theorem is given in Section A.6 in Appendix A.

Property (1) states that the structured lower approximation contains the structured Pawlak lower approximation. Property (2) states that the structured lower approximation is consistent with Pawlak lower approximation induced by all attributes in the sense that both of them cover the same set of objects.

Similar connections can be established between $\underline{sapr}(X)$ and the adaptive lower approximation.

Theorem 7. *The following properties hold:*

$$\begin{aligned} &(1) \qquad \text{For any sequence } Seq_n \text{ of sets of attributes, } \underline{dapr}_n(X) \subseteq \underline{sapr}(X), \\ &(2) \qquad \text{For } Seq_{|AT|} = \{A_1, A_2, \cdots, AT\}, \; \bigcup \underline{dapr}_{|AT|}(X) = \bigcup \underline{sapr}(X). \end{aligned} \tag{4.3}$$

The proof of this theorem is given in Section A.7 in Appendix A.

Property (1) states that the structured lower approximation contains the adaptive approximation. Property (2) states that the structured lower approximation covers the same set of objects with the adaptive lower approximation considering all attributes.

From Theorems 6 and 7, it can be concluded that by considering all attributes, all these four kinds of lower approximations (i.e., $\underline{apr}(X), \underline{bapr}(X), \underline{dapr}(X)$ and $\underline{sapr}(X)$) actually cover the same set of objects, that is, they are consistent.

One difference between the structured lower approximation and the other three is that $\underline{sapr}(X)$ is a covering-based formulation while the others are partition-based. In other words, overlap between components is allowed in $\underline{sapr}(X)$ while it is not in the other three.

Another difference is that the structured lower approximation always retains maximal conjunctively definable subsets of X which may give general rules which are preferred in concept analysis. This is one of the most important advantages of the structured rough set approximations.

4.5 Reduced Structured Rough Set Approximations

In the structured Pawlak lower approximation $\underline{bapr}(X)$ and the adaptive lower approximation $\underline{dapr}(X)$, for any proper subsets $\mathbb{B} \subsetneq \underline{bapr}(X)$ and $\mathbb{D} \subsetneq \underline{dapr}(X)$, we have:

$$\begin{array}{ll} (1) & \bigcup \mathbb{B} \neq \bigcup \underline{bapr}(X), \\ (2) & \bigcup \mathbb{D} \neq \bigcup \underline{dapr}(X). \end{array} \tag{4.4}$$

That is, to cover all objects in the approximations, we cannot remove any equivalence classes. For the structured lower approximation $\underline{sapr}(X)$, it is possible to remove some description pairs in $\underline{sapr}(X)$ and, at the same time, to cover the same objects. We may find a proper subset $\mathbb{R} \subsetneq \underline{sapr}(X)$ such that:

$$\bigcup \mathbb{R} = \bigcup \underline{sapr}(X). \tag{4.5}$$

This suggests that the structured lower approximation contains redundant description pairs. By removing those redundant description pairs, we may obtain a reduced structured lower approximation.

In this thesis, two methods are presented to reduce the redundancy. One is to find a $\cup$-reduct of $\underline{sapr}(X)$, which is a reduced structured lower approximation. The other one is to find a $\cup$-reduct of all extension-based maximal elements in $\underline{sapr}(X)$, which is a reduced structured lower approximation with extension-based maximal elements.

4.5.1 Reduced Structured Lower Approximations

The notion of $\cup$-reduct (or union-reduct) is proposed by Pawlak [26, 44] to define reducts of a decision rule.

Definition 17. *Suppose $\mathbb{S}$ is a family of subsets of U. A subset $\mathbb{R} \subseteq \mathbb{S}$ is called a union-reduct or $\cup$-reduct of $\mathbb{S}$ if it satisfies the following two conditions:*

$$\begin{aligned} &(1) \qquad \bigcup \mathbb{R} = \bigcup \mathbb{S}, \\ &(2) \qquad \forall\, S \in \mathbb{R},\ \bigcup (\mathbb{R} - \{S\}) \neq \bigcup \mathbb{S}. \end{aligned} \tag{4.6}$$

Semantically, the first condition can be summarized as "jointly sufficient," meaning that sets in a $\cup$-reduct $\mathbb{R}$ can jointly cover the same set of objects as the original family $\mathbb{S}$. The second condition can be summarized as "individually necessary," meaning that each set in a $\cup$-reduct $\mathbb{R}$ is necessary and cannot be removed. In other words, a $\cup$-reduct is a minimal family of sets in $\mathbb{S}$ that covers the same objects as $\mathbb{S}$. There may exist more than one $\cup$-reduct of $\mathbb{S}$.

The notion of $\cup$-reduct can be applied to a covering. One special kind of elements in a covering is called the $\cup$-irreducible elements. Such an element can not be expressed by the union of other elements in the covering. A $\cup$-reduct of a covering is different from the set of all $\cup$-irreducible elements of a covering. The latter is mistakenly called a $\cup$-reduct of a covering by some researchers [48, 54].

Definition 18. *A $\cup$-reduct of a structured lower approximation is called a reduced structured lower approximation.*

Since a $\cup$-reduct is not unique, several possible reduced structured lower approximations may be derived for one structured lower approximation. Let $R_{\cup}(sapr(X))$ denote the family of all reduced structured lower approximations of a subset $X \subseteq U$. One can establish the following relationships between $R_{\cup}(sapr(X))$ and different kinds of lower approximations.

Theorem 8. *The following properties hold for $R_{\cup}(sapr(X))$:*

$$
\begin{aligned}
&(1) \quad && \forall \mathbb{R} \in R_{\cup}(sapr(X)), \\
& && \bigcup \mathbb{R} = \bigcup \underline{sapr}(X) = \underline{apr}_{E_{AT}}(X) = \bigcup \underline{bapr}_{E_{AT}}(X) = \bigcup \underline{dapr}_{|AT|}(X), \\
&(2) && \underline{bapr}_{E_{AT}}(X) \in R_{\cup}(sapr(X)), \\
&(3) && \underline{dapr}_{|AT|}(X) \in R_{\cup}(sapr(X)).
\end{aligned}
\tag{4.7}
$$

The proof of this theorem is given in Section A.8 in Appendix A.

Property (1) states that all $\cup$-reducts of $\underline{sapr}(X)$ and all the four different kinds of lower approximations, that is, unstructured and structured Pawlak lower approximation, adaptive lower approximation and structured lower approximation, are equivalent in the sense that they cover the same subset of X.

Properties (2) and (3) state that when all attributes are considered, both the structured Pawlak lower approximation and the adaptive lower approximation are reduced structured lower approximations. This also infers that the structured lower approximation provides a general formulation of approximations with structural information.

Although redundancy in $\underline{sapr}(X)$ can be reduced by finding its $\cup$-reducts, there are several disadvantages of this method. One disadvantage concerns classification rules. When constructing classification rules, one prefers to get more general rules which are created by a large definable set containing as many objects as possible. On the other hand, $\underline{bapr}(X)$ and $\underline{dapr}(X)$ contain equivalence classes which are the minimal nonempty conjunctively definable sets, that is, no proper nonempty subset of an equivalence class is conjunctively definable. A $\cup$-reduct of the structured lower approximation may not necessarily use larger conjunctively definable subsets of X. In other words, a $\cup$-reduct of $\underline{sapr}(X)$ may contain a family of smaller conjunctively definable sets whose union is a larger one. This problem will be solved in the second method by introducing an extra step before constructing a $\cup$-reduct.

Example 11. *We give several reduced structured lower approximations of the structured lower approximation constructed in Section 4.3 as follows:*

$$\begin{aligned}
\mathbb{R}_1 &= \big\{\{o_1\},\{o_2\},\{o_3\},\{o_4\}\big\},\\
\mathbb{R}_2 &= \big\{\{o_1,o_2,o_3\},\{o_4\}\big\},\\
\mathbb{R}_3 &= \big\{\{o_1,o_2,o_3\},\{o_1,o_2,o_4\}\big\}.
\end{aligned}$$

From these examples, we can observe the following facts:

(1) There may be more than one $\cup$-reduct of a structured lower approximation.

(2) $\mathbb{R}_1$ is the structured Pawlak lower approximation constructed in Section 3.1.4.

(3) $\mathbb{R}_2$ is the adaptive lower approximation constructed in Section 3.2.4.

(4) By comparing $\mathbb{R}_3$ and the other two $\cup$-reducts, we can see the disadvantage of this method for reducing the redundancy. That is, a family of smaller sets may be preserved instead of a larger one. For example, $\{o_1, o_2, o_4\}$ is kept in $\mathbb{R}_3$, while sets $\{o_1\}, \{o_2\}, \{o_4\}$ are kept instead in $\mathbb{R}_1$.

4.5.2 Reduced Structured Lower Approximations with Extension-Based Maximal Elements

As was mentioned previously, one main disadvantage of reducing redundancy in $\underline{sapr}(X)$ by finding a $\cup$-reduct is that a set of smaller conjunctively definable sets may be preserved instead of a larger one. In this section, a second method is proposed to solve this problem and reduce the redundancy in $\underline{sapr}(X)$ by introducing an extra step to find all extension-based maximal elements in $\underline{sapr}(X)$ before constructing a $\cup$-reduct.

Definition 19. *Given a family of sets $\mathbb{S}$, a set $S_M \in \mathbb{S}$ is an extension-based maximal element in $\mathbb{S}$ if there does not exist a set $S \in \mathbb{S}$ such that $S_M \subsetneq S$.*

Let $\mathbb{M}_{ex}(\underline{sapr}(X))$ denote the family of all extension-based maximal elements in $\underline{sapr}(X)$. The family $\mathbb{M}_{ex}(\underline{sapr}(X))$ can be easily obtained by removing all $S \in \underline{sapr}(X)$ such that there exists another set $S' \in \underline{sapr}(X)$ with $S \subsetneq S'$.

Theorem 9. *The following properties hold:*

$$\begin{aligned} &(1) \qquad \mathbb{M}_{ex}(\underline{sapr}(X)) \;\subseteq\; \underline{sapr}(X), \\ &(2) \qquad \bigcup \mathbb{M}_{ex}(\underline{sapr}(X)) \;=\; \bigcup \underline{sapr}(X). \end{aligned} \tag{4.8}$$

That is, $\mathbb{M}_{ex}(\underline{sapr}(X))$ is a subset of $\underline{sapr}(X)$, which covers the same set of objects as $\underline{sapr}(X)$.

The proof is given in Section A.9 of Appendix A.

A $\cup$-reduct of $\mathbb{M}_{ex}(\underline{sapr}(X))$ immediately leads to a reduced structured lower approximation with extension-based maximal elements.

Definition 20. *A $\cup$-reduct of $\mathbb{M}_{ex}(\underline{sapr}(X))$ is called a reduced structured lower approximation with extension-based maximal elements.*

The reduced structured lower approximations with extension-based maximal elements are not unique.

An advantage of this method is that not only is the redundancy in $\underline{sapr}(X)$ reduced, but also the general conjunctively definable sets, which may give general classification rules, are kept.

By definitions, we can establish the following set-inclusion relationship.

Theorem 10. *Let $R_{\cup}(\cdot)$ denote the family of all $\cup$-reducts. The following property holds:*

$$R_{\cup}(\mathbb{M}_{ex}(\underline{sapr}(X))) \subseteq R_{\cup}(\underline{sapr}(X)). \tag{4.9}$$

The proof of this theorem is given in Section A.10 in Appendix A.

It may happen that the structured Pawlak lower approximation $\underline{bapr}(X)$ and adaptive lower approximation $\underline{dapr}(X)$, although members of $R_{\cup}(\underline{sapr}(X))$, are not members of $R_{\cup}(\mathbb{M}_{ex}(\underline{sapr}(X)))$. This stems from the fact that some equivalence classes in $\underline{bapr}(X)$ and $\underline{dapr}(X)$ may not be extension-based maximal elements in $\underline{sapr}(X)$.

Example 12. *The structured lower approximation constructed in Section 4.3 is used as an example to illustrate the computation of a reduced structured lower approximation with extension-based maximal elements. Recall that the structured lower approximation is given by Table 4.2. There are two steps as follows:*

Step 1: *Construct the set of all extension-based maximal elements:*

$$\mathbb{M}_{ex}(\underline{sapr}(X)) \;=\; \big\{\{o_1,o_2,o_3\},\{o_1,o_2,o_4\},\{o_1,o_3,o_4\}\big\}.$$

Step 2: *Find a* $\cup$*-reduct of* $\mathbb{M}_{ex}(\underline{sapr}(X))$*. For this example, we can find three* $\cup$*-reducts of* $\mathbb{M}_{ex}(\underline{sapr}(X))$*:*

$$\begin{aligned}
\mathbb{R}_{ex1} &= \big\{\{o_1,o_2,o_3\},\{o_1,o_2,o_4\}\big\},\\
\mathbb{R}_{ex2} &= \big\{\{o_1,o_2,o_3\},\{o_1,o_3,o_4\}\big\},\\
\mathbb{R}_{ex3} &= \big\{\{o_1,o_2,o_4\},\{o_1,o_3,o_4\}\big\}.
\end{aligned}$$

4.6 Simplifying Classification Rules by Using Structured Approximations

In rough set theory, classification rules are constructed from approximations. Based on a lower approximation, certain rules (i.e., rules without any uncertainty) can be created. This section presents brief comparisons of classification rules learned by the four kinds of lower approximations by using the lower approximations constructed in the previous examples. These rules demonstrate that by reducing redundancy, the structured lower approximation may give the simpler or more general classification rules.

4.6.1 Classification Rules Learned by Pawlak Approximations

In either structured or unstructured Pawlak lower approximations, each equivalence class gives a classification rule. An equivalence class $[x]_{E_A}$ can be described or defined by the following formula:

$$\bigwedge_{a \in A} (a = I_a(x)). \tag{4.10}$$

Therefore, for each equivalence class $[x]_{E_A} \in \underline{bapr}(X)$ or $[x]_{E_A} \subseteq \underline{apr}(X)$, the corresponding classification rule is created as:

$$\bigwedge_{a \in A} (a = I_a(x)) \quad \Longrightarrow \quad \text{class of } X. \tag{4.11}$$

It means that if an object satisfies the formula in the left-hand side of the rule, then this object belongs to the same class as the set X.

Example 13. *Since the structured and unstructured Pawlak lower approximations cover the same set of objects and the structured one explicitly displays the equivalence classes used, the structured one is used to construct the classification rules here. Recall that in Section 3.1.4, the structured Pawlak lower approximation constructed by using all attributes AT is:*

$$\begin{aligned} \underline{bapr}_3(X) &= \{K \in U/E_{\{\text{Height},\text{Hair},\text{Eyes}\}} \mid K \subseteq X\} \\ &= \big\{\{o_1\}, \{o_2\}, \{o_3\}, \{o_4\}\big\}. \end{aligned}$$

Therefore, we can learn four rules from the four equivalence classes contained.

Rule 1: (derived by $\{o_1\}$)

(Height=short) $\wedge$ (Hair=blond) $\wedge$ (Eyes=blue) $\Longrightarrow$ *class of X;*

Rule 2: (derived by $\{o_2\}$)

(Height=short) $\wedge$ (Hair=blond) $\wedge$ (Eyes=brown) $\Longrightarrow$ *class of X;*

Rule 3: (derived by $\{o_3\}$)

(Height=short) $\wedge$ (Hair=red) $\wedge$ (Eyes=blue) $\Longrightarrow$ *class of X;*

Rule 4: *(derived by* $\{o_4\}$*)*

$$(\text{Height=tall}) \wedge (\text{Hair=blond}) \wedge (\text{Eyes=blue}) \quad \Longrightarrow \quad \textit{class of } X.$$

The rules created by Pawlak approximations are complicated since all attributes are used in each single rule.

4.6.2 Classification Rules Learned by Adaptive Approximations

Similar to Pawlak approximations, each equivalence class in an adaptive lower approximation can give a classification rule. Suppose the sequence of attributes is $Seq_n = \{A_1, A_2, \cdots, A_n\}$. An equivalence class $[x]_{E_{A_i}} \in U_i/E_{A_i}$ $(1 \leq i \leq n)$ creates the following rule:

$$\bigwedge_{a \in A_i} \big(a = I_a(x)\big) \quad \Longrightarrow \quad \text{class of } X. \tag{4.12}$$

Thus, each individual rule will use at most $|A_n|$ attributes.

Example 14. *The adaptive lower approximation constructed in Section 3.2.4 is:*

$$\underline{dapr}_3(X) \;=\; \big\{\{o_1, o_2, o_3\}, \{o_4\}\big\}.$$

For the equivalence classes contained, $\{o_1, o_2, o_3\} \in X_1/E_{\{\text{Height}\}}$, $\{o_4\} \in X_2/E_{\{\text{Height,Hair}\}}$. *Therefore, two rules can be obtained correspondingly.*

Rule 1: *(derived by* $\{o_1, o_2, o_3\}$*)*

$$\text{Height=short} \quad \Longrightarrow \quad \textit{class of } X;$$

Rule 2: *(derived by* $\{o_4\}$*)*

$$(\text{Height=tall}) \wedge (\text{Hair=blond}) \quad \Longrightarrow \quad \textit{class of } X.$$

Compared with the rules in Example 13, these rules learned by the adaptive lower approximation are simplified in two ways:

(1) Less total number of rules are created;

(2) Each individual rule is simplified by using less attributes.

4.6.3 Classification Rules Learned by Structured Approximations

For structured approximations, classification rules are given by either a reduced structured lower approximation or a reduced structured lower approximation with extension-based maximal elements. The formula of a conjunctively definable set in the approximation immediately gives the left-hand side of a classification rule. Suppose a conjunctive formula is:

$$(a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_n = v_n). \tag{4.13}$$

The corresponding rule is:

$$(a_1 = v_1) \wedge (a_2 = v_2) \wedge \cdots \wedge (a_n = v_n) \quad \Longrightarrow \quad \text{class of } X. \tag{4.14}$$

For those conjunctively definable sets that have more than one conjunctive formula, one can choose one from the formulas according to specific heuristics given by users. Two possible heuristics may be:

(1) Choosing formulas using the least number of attributes;

(2) Choosing formulas using the largest number of attributes appearing in other rules.

Example 15. *In Examples 11 and 12, we constructed three reduced structured lower approximations $\mathbb{R}_1, \mathbb{R}_2$ and $\mathbb{R}_3$, and three reduced structured lower approximations with extension-based maximal elements $\mathbb{R}_{ex1}, \mathbb{R}_{ex2}$ and $\mathbb{R}_{ex3}$. For one conjunctively definable set, there may be more than one conjunctive formulas with the least number* *of attributes. Among these formulas, one formula is randomly chosen in this example. The following presents the classification rules created by some of them.*

(1) Classification rules learned by $\mathbb{R}_1$*:*

Rule 1: *(derived by* $\{o_1\}$*)*

(Height=short) $\wedge$ (Hair=blond) $\wedge$ (Eyes=blue) $\Longrightarrow$ *class of* X*;*

Rule 2: *(derived by* $\{o_2\}$*)*

(Height=short) $\wedge$ (Eyes=brown) $\Longrightarrow$ *class of* X*;*

Rule 3: *(derived by* $\{o_3\}$*)*

(Height=short) $\wedge$ (Hair=red) $\Longrightarrow$ *class of* X*;*

Rule 4: *(derived by* $\{o_4\}$*)*

(Height=tall) $\wedge$ (Hair=blond) $\Longrightarrow$ *class of* X*.*

(2) Classification rules learned by $\mathbb{R}_2$*:*

Rule 1: *(derived by* $\{o_1, o_2, o_3\}$*)*

Height=short $\Longrightarrow$ *class of* X*;*

Rule 2: *(derived by* $\{o_4\}$*)*

(Height=tall) $\wedge$ (Eyes=blue) $\Longrightarrow$ *class of* X*.*

(3) Classification rules learned by $\mathbb{R}_3 = \mathbb{R}_{ex1}$*:*

Rule 1: *(derived by* $\{o_1, o_2, o_3\}$*)*

Height=short $\Longrightarrow$ *class of* X*;*

Rule 2: *(derived by* $\{o_1, o_2, o_4\}$*)*

Hair=blond $\Longrightarrow$ *class of* X*.*

Although $\mathbb{R}_1$ and $\mathbb{R}_2$ are the same with structured Pawlak lower approximation and adaptive lower approximation, respectively, the rules are simplified by using fewer attributes. By introducing extension-based maximal elements, general sets are preserved which may give general rules.

By comparing all the classification rules created by $\underline{bapr}(X)$, $\underline{dapr}(X)$ and $\underline{sapr}(X)$, the structured lower approximation may simplify rules in the following two ways:

(1) Reduce individual rules:

By definition of a structured lower approximation, most general subsets are all preserved, which may give general rules. Moreover, by introducing extension-based maximal elements to reduce a structured lower approximation, larger sets are preferred and general rules may be created accordingly.

(2) Reduce total number of rules:

By adopting $\cup$-reduct to reduce a structured lower approximation, the total number of rules may be reduced.

4.7 Relationships to LERS Systems

With reference to the classical view of concepts, the formulation of Grzymała-Busse [9, 10] in the LERS (Learning from Examples based on Rough Sets) systems and our formulation are complementary to each other, focusing on the intension and extension of a concept, respectively. In LERS, Grzymała-Busse considers the set-inclusion relation $\subseteq$ between sets of atomic formulas, reflecting the complexity or generality of the intensions of conjunctive concepts, but he does not consider the set-inclusion relationship between conjunctively definable sets, namely, extensions of conjunctive concepts, defined by sets of atomic formulas. In our formulation with extension-based maximal elements, we consider the set-inclusion relation $\subseteq$ between conjunctively definable sets, but we do not consider the complexity or generality of conjunctive formulas used in defining conjunctively definable sets.

Our formulation	**Grzymała-Busse formulation**
atomic formula $a = v$	attribute-value pair $t = (a, v)$
meaning set $m(a = v)$	block $[(a, v)]$
conjunctive formula p	set T of attribute-value pairs (corresponding to atomic formulas in p)
extension-based maximal elements $\mathbb{M}_{ex}(\underline{sapr}(X))$ in $\underline{sapr}(X)$	minimal complex of X
$\cup$-reduct $\mathbb{R}$ of $\mathbb{M}_{ex}(\underline{sapr}(X))$	local covering $\mathbb{T}$ of $\underline{apr}(X)$

Table 4.3: Correspondences between Our and Grzymała-Busse's Formulations

Table 4.3 gives some relevant correspondences between notions of the two formulations. An attribute-value pair (a, v) corresponds to an atomic formula $a = v$. The block of an attribute-value pair $t = (a, v)$, denoted $[t]$, is defined as the set of all objects that take value v on attribute a which is the meaning set of atomic formula $a = v$. That is, $[(a, v)] = \{x \in U \mid I_a(x) = v\} = m(a = v)$. A set T of attribute-value pairs corresponds to the set of all atomic formulas in a formula of DL, and the block of T, denoted $[T]$, is the meaning set of the corresponding formula. In other words, T is another representation of the intension of a conjunctively definable set. A set T of attribute-value pairs is a minimal complex of $\underline{apr}(X)$ if it satisfies the conditions:

$$\begin{aligned} &(1) \qquad \emptyset \neq [T] \;=\; \bigcap_{t \in T} [t] \;\subseteq\; \underline{apr}(X), \\ &(2) \qquad \forall\, T' \subsetneq T,\; [T'] \;=\; \bigcap_{t \in T'} [t] \;\not\subseteq\; \underline{apr}(X). \end{aligned} \tag{4.15}$$

In other words, Grzymała-Busse considers the intensions of concepts by using minimal sets of attribute-value pairs; we consider the extensions of concepts by using maximal conjunctively definable sets. A family $\mathbb{T}$ of sets of attribute-value pairs is called a local covering of $\underline{apr}(X)$ if and only if it satisfies the conditions:

(1) each member T of $\mathbb{T}$ is a minimal complex of $\underline{apr}(X)$,

(2) $\bigcup_{T \in \mathbb{T}} [T] = \underline{apr}(X)$,

(3) $\mathbb{T}$ is minimal, i.e., $\mathbb{T}$ has the smallest possible number of members.

A local covering is a family of sets of attribute-value pairs and a $\cup$-reduct of $\mathbb{M}(\underline{sapr}(X))$ is a family of maximal conjunctively definable sets. That is, from the viewpoint of intension, Grzymała-Busse uses a family of minimal complexes whose blocks are subsets of X; from the viewpoint of extension, we use a family of maximal conjunctively definable subsets of X.

4.8 Summary

This chapter presents the main results of structured rough set approximations. A generalized definition of structured rough set lower approximations is given, which uses the family of conjunctively definable sets as building blocks of the approximations. We adopt the notion of $\cup$-reduct and present two methods to reduce the redundancy in a structured lower approximation. One is to directly find a $\cup$-reduct of a structured lower approximation, while the other is to find a $\cup$-reduct of extension-based maximal elements in a structured lower approximation.

This chapter also investigates the relationships between different rough set lower approximations with structural information, that is, structured lower approximations, structured Pawlak lower approximations and adaptive lower approximations. All these lower approximations are equivalent in the sense that they cover the same set of objects. However, they are quite different with regard to the structural information. Each component in the structured lower approximation has an explicit and well-defined description, which can benefit the rule learning process in concept analysis with rough set theory by directly giving the semantics of a rule. Examples are presented in this chapter to show that simplified rules could be learned by structured lower approximations.

While Pawlak and adaptive approximations are partition-based, structured approximations are covering-based. This chapter discusses the relationships between structured approximations and another covering-based system LERS.

Chapter 5

CONCLUSIONS AND FUTURE WORK

This chapter summarizes this thesis and discusses several possible directions for future work.

5.1 Conclusions

This thesis presents a new formulation of rough set approximations, the structured rough set approximations, which considers the structural information. The proposed structured rough set approximations may simplify classification rules.

There are several existing formulations of rough set approximations that preserve structural information. The structured Pawlak approximations use equivalence classes as building blocks, which are actually the minimal definable sets with respect to a specific set of attributes. Although one can find a formula or a family of sets of attribute-value pairs to explain the semantics of each equivalence class, all attributes considered will be used in the formula. Since each equivalence class in a structured Pawlak approximation induces a classification rule, all attributes considered will be contained in each individual rule. Such rules may be lengthy and are not preferred in concept analysis.

The adaptive approximations can capture better structural information by using a sequence of equivalence classes with different granularities, instead of just one granularity as is used in Pawlak approximations. It can be equivalently considered as combining the small equivalence classes in Pawlak approximations into larger equivalence classes. By using large equivalence classes, more general rules with less attributes can be learned. However, the adaptive approximations are also confined by the partition-based approach. Small equivalence classes are necessary in order to get a precise approximation. However, it is still possible to expand components in an adaptive approximation into a larger definable set by combining them. Such a combination will lead to overlap between components, which is not allowed in a partition-based approach.

The proposed structured rough set approximations use the family of all conjunctively definable sets as building blocks. This makes the structured approximations a covering-based formulation. Moreover, we give the new framework followed by the proposed structured approximations. This new framework is semantically sound since we adopt the description language to formally and explicitly represent the semantics. Compared with the Pawlak framework which does not have explicit explanation of the semantics, the new framework may be more advantageous from the view of semantical issues.

The proposed structured approximations are consistent with the existing formulations since the same sets of objects are covered. However, there may be redundancy in a structured approximation and one can use various algorithms to reduce the redundancy. In this thesis, the notion of $\cup$-reduct is adopted and two methods to reduce the redundancy are presented.

Besides simplifying classification rules, one important contribution of the proposed structured rough set approximations concerns semantics. On the one hand, a clearly represented description of each conjunctively definable set in the structured approximation leads to a well-defined semantical explanation of the whole approximation. On the other hand, as a covering-based formulation, the structured approximation brings a semantically sound model to covering-based rough set theory, while most existing covering-based frameworks have difficulties explaining the semantics of patches or components in a covering.

5.2 Future Work

A possible direction of future work is to generalize our new framework given by Figure 4.1 to deal with incomplete information.

In a real world application, although each object can actually take one value from the corresponding domain of each attribute, one may not be able to get this exact value due to uncertain or incomplete information. Therefore, there are two kinds of information tables, namely, complete and incomplete information tables.

In this thesis, we only consider the case of complete information tables, in which each object can only take one value on each attribute. Recall that the definition of such a complete information table is described by a tuple:

$$T = (U, AT, \{V_a \mid a \in AT\}, \{I_a : U \to V_a \mid a \in AT\}). \tag{5.1}$$

When there is incomplete information, an object may be known to take some value from a set of possible values. Therefore, an incomplete information table [15,16,40,46] is formulated as:

$$\widetilde{T} = (U, AT, \{V_a \mid a \in AT\}, \{\widetilde{I}_a : U \to 2^{V_a} - \{\emptyset\} \mid a \in AT\}). \tag{5.2}$$

While the description function $I_a : U \to V_a$ in a complete information table assigns exactly one value from V_a for each object, the description function $\widetilde{I}_a : U \rightarrow 2^{V_a} - \{\emptyset\}$ in an incomplete information table assigns a nonempty subset of V_a for each object. Among values in this assigned subset, the object actually takes only one value, although we do not know what it is due to incomplete information.

Most research [12,13,23,30,33,35] concerning incomplete information in rough set theory tries to generalize Pawlak framework given by Figure 4.1 (a). Basically, they give generalizations of equivalence relation in a complete information table, such as tolerance relation, non-symmetric similarity relation, valued tolerance relation and limited tolerance relation. Based on such relations, the corresponding generalized frameworks with incomplete information can be derived following the steps in the Pawlak framework. However, all these generalized relations have their own limitations [35].

By noticing that our new framework in Figure 4.1 (b) does not need to define a relation first, a generalization of our framework with incomplete information may overcome those limitations caused by generalized relations.

Lipski [15,16] proposed to interpret an incomplete information table as a family of complete information tables. A complete information table T is called a completion of an incomplete information table $\widetilde{T}$ if it satisfies the condition:

$$\forall\, x \in U,\ \forall\, a \in AT,\ I_a(x) \in \widetilde{I}_a(x), \tag{5.3}$$

where $I_a(\cdot)$ and $\widetilde{I}_a(\cdot)$ denote the description functions in table T and $\widetilde{T}$, respectively. Let $\mathrm{COMP}(\widetilde{T})$ denote the family of all completions of an incomplete information table $\widetilde{T}$. Each table in $\mathrm{COMP}(\widetilde{T})$ may be the real information table.

In an incomplete information table $\widetilde{T}$, for a formula p in the description language

DL, one can obtain two bounds of its real meaning set:

$$
\begin{aligned}
m_*(p) &= \{x \in U \mid \forall T \in \mathrm{COMP}(\widetilde{T}),\ x \in m^T(p)\} \\
&= \bigcap_{T \in \mathrm{COMP}(\widetilde{T})} m^T(p), \\
m^*(p) &= \{x \in U \mid \exists T \in \mathrm{COMP}(\widetilde{T}),\ x \in m^T(p)\} \\
&= \bigcup_{T \in \mathrm{COMP}(\widetilde{T})} m^T(p).
\end{aligned} \tag{5.4}
$$

$m^T(p)$ denotes the meaning set of p in a complete table T. The lower bound $m_*(p)$ actually includes objects which definitely satisfy p, and the upper bound $m^*(p)$ includes objects which possibly satisfy p. Let $m(p)$ denote the real meaning set, then it can be concluded that $m_*(p) \subseteq m(p) \subseteq m^*(p)$.

Therefore, instead of one family of conjunctively definable sets in a complete table, one can obtain two families of conjunctively definable sets in an incomplete table:

$$
\begin{aligned}
\mathrm{CDEF}_*(U) &= \{m_*(p) \mid p \text{ is a conjunctive formula in } DL\}, \\
\mathrm{CDEF}^*(U) &= \{m^*(p) \mid p \text{ is a conjunctive formula in } DL\}.
\end{aligned} \tag{5.5}
$$

Following the steps in our new framework, a generalized rough set model with incomplete information can be established without using a relation.

Another possible direction for future work may involve taking intension into consideration to reduce a structured lower approximation. This thesis presents the reduced structured lower approximations with extension-based maximal elements. With this method, individual rules induced will cover more objects and are considered more general. However, such rules do not necessarily use fewer attributes. There may exist smaller conjunctively definable sets that can be defined by conjunctive formulas with fewer attributes. This may happen when a small group of objects in an information table takes special values on a small number of attributes. One can solve this problem by considering intension, that is, formulas in the new framework, instead of extension to reduce a structured lower approximation.

Other possible directions for future work may include investigating algorithms or methods to reduce redundancy in a structured rough set approximation other than using a $\cup$-reduct.

Appendix A

Proof of Theorems

Detailed proofs of all theorems in this thesis are given below.

A.1 Proof of Theorem 1

Theorem 1 *Pawlak and Bryniarski approximations can define each other as follows:*

$$\begin{aligned} \underline{apr}(X) &= \bigcup \underline{bapr}(X), \\ \overline{apr}(X) &= \bigcup \overline{bapr}(X), \end{aligned} \tag{A.1}$$

and

$$\begin{aligned} \underline{bapr}(X) &= \{[x] \in U/E \mid [x] \subseteq \underline{apr}(X)\}, \\ \overline{bapr}(X) &= \{[x] \in U/E \mid [x] \subseteq \overline{apr}(X)\}. \end{aligned} \tag{A.2}$$

Proof.

Equation (A.1) can be directly verified by the definitions of Pawlak and Bryniarski approximations (i.e., Definitions 2 and 3). The proof of Equation (A.2) is given as follows.

1. Prove that $\underline{bapr}(X) = \{[x] \in U/E \mid [x] \subseteq \underline{apr}(X)\}$.

   According to set theory, it can be proved by verifying the following two subset relationships:

   (i) $\underline{bapr}(X) \subseteq \{[x] \in U/E \mid [x] \subseteq \underline{apr}(X)\}$:

   According to the definition of Bryniarski approximations (i.e., Definition 3), for any equivalence class $[x]$ in $\underline{bapr}(X)$, we have $[x] \subseteq X$. Since any object y in $[x]$ has the same equivalence class with x, that is, $[y] = [x]$, it follows that $[y] \subseteq X$. This infers that $y \in \underline{apr}(X)$. Thus, all objects in $[x]$ are also in $\underline{apr}(X)$, that is, $[x] \subseteq \underline{apr}(X)$. So it can be concluded that $\underline{bapr}(X) \subseteq \{[x] \in U/E \mid [x] \subseteq \underline{apr}(X)\}$.

   (ii) $\underline{bapr}(X) \supseteq \{[x] \in U/E \mid [x] \subseteq \underline{apr}(X)\}$:

   Since an equivalence relation is reflexive, that is, $x \in [x]$, for any equivalence class $[x] \subseteq \underline{apr}(X)$, the object x is in $\underline{apr}(X)$. According to the definition of Pawlak approximations (i.e., Definition 2), $[x] \subseteq X$. This infers that $[x] \in \underline{bapr}(X)$. So it can be concluded that $\underline{bapr}(X) \supseteq \{[x] \in U/E \mid [x] \subseteq \underline{apr}(X)\}$.

2. Prove that $\overline{bapr}(X) = \{[x] \in U/E \mid [x] \subseteq \overline{apr}(X)\}$.

   According to set theory, it can be proved by verifying the following two subset relationships:

   (i) $\overline{bapr}(X) \subseteq \{[x] \in U/E \mid [x] \subseteq \overline{apr}(X)\}$:

   According to the definition of Bryniarski approximations (i.e., Definition 3), for any equivalence class $[x]$ in $\overline{bapr}(X)$, we have $[x] \cap X \neq \emptyset$. Since any object y in $[x]$ has the same equivalence class with x, that is, $[y] = [x]$, it follows that $[y] \cap X \neq \emptyset$. This infers that $y \in \overline{apr}(X)$. Thus, all objects in $[x]$ are also in $\overline{apr}(X)$, that is, $[x] \subseteq \overline{apr}(X)$. So it can be concluded that $\overline{bapr}(X) \subseteq \{[x] \in U/E \mid [x] \subseteq \overline{apr}(X)\}$.

(ii) $\overline{bapr}(X) \supseteq \{[x] \in U/E \mid [x] \subseteq \overline{apr}(X)\}$:

Since an equivalence relation is reflexive, that is, $x \in [x]$, for any equivalence class $[x] \subseteq \overline{apr}(X)$, the object x is in $\overline{apr}(X)$. According to the definition of Pawlak approximations (i.e., Definition 2), $[x] \cap X \neq \emptyset$. This infers that $[x] \in \overline{bapr}(X)$. So it can be concluded that $\overline{bapr}(X) \supseteq \{[x] \in U/E \mid [x] \subseteq \overline{apr}(X)\}$.

□

A.2 Proof of Theorem 2

Theorem 2 *The meaning set of a formula in DL can be computed as:*

$$\begin{aligned} &(1) \qquad m(a = v) = \{x \in U \mid I_a(x) = v\}, \\ &(2) \qquad m(p \wedge q) = m(p) \cap m(q), \\ &(3) \qquad m(p \vee q) = m(p) \cup m(q). \end{aligned} \tag{A.3}$$

Proof.

(1) Prove that $m(a = v) = \{x \in U \mid I_a(x) = v\}$.

It can be directly verified by the definition of satisfiability (i.e., Definition 6), that is, $x \models (a = v)$, if $I_a(x) = v$.

(2) Prove that $m(p \wedge q) = m(p) \cap m(q)$.

According to the definition of meaning sets and satisfiability of formulas (i.e., Definitions 7 and 6), the meaning set $m(p \wedge q)$ can be transformed into $m(p) \cap m(q)$

in the following way:

$$\begin{aligned} m(p \wedge q) &= \{x \in U \mid x \models (p \wedge q)\} \\ &= \{x \in U \mid (x \models p) \wedge (x \models q)\} \\ &= \{x \in U \mid x \models p\} \cap \{x \in U \mid x \models q\} \\ &= m(p) \cap m(q). \end{aligned} \tag{A.4}$$

(3) Prove that $m(p \vee q) = m(p) \cup m(q)$.

According to the definition of meaning sets and satisfiability of formulas (i.e., Definitions 7 and 6), the meaning set $m(p \vee q)$ can be transformed into $m(p) \cup m(q)$ in the following way:

$$\begin{aligned} m(p \vee q) &= \{x \in U \mid x \models (p \vee q)\} \\ &= \{x \in U \mid (x \models p) \vee (x \models q)\} \\ &= \{x \in U \mid x \models p\} \cup \{x \in U \mid x \models q\} \\ &= m(p) \cup m(q). \end{aligned} \tag{A.5}$$

$\square$

A.3 Proof of Theorem 3

Theorem 3 *Suppose* $\{S_1, S_2, \cdots, S_n\}$ *is a family of sets of attribute-value pairs where* $S_i \subseteq AV$ $(1 \leq i \leq n)$, *one can compute its meaning set by using set intersection and* *set union as:*

$$\begin{aligned} m(\{S_1, S_2, \cdots, S_n\}) &= \bigcup_{1 \le i \le n} m(\{S_i\}) \\ &= \bigcup_{1 \le i \le n} \bigcap_{(a,v) \in S_i} \{x \mid x \in U,\ x \models (a, v)\}. \end{aligned} \tag{A.6}$$

Proof.

According to the definition of meaning sets and satisfiability of attribute-value pairs (i.e., Definitions 9 and 8), we transform the meaning set $m(\{S_1, S_2, \cdots, S_n\})$ and prove the theorem as follows.

$$\begin{aligned} m(\{S_1, S_2, \cdots, S_n\}) &= \{x \in U \mid x \models \{S_1, S_2, \cdots, S_n\}\} \\ &= \{x \in U \mid (x \models S_1) \vee (x \models S_2) \vee \cdots \vee (x \models S_n)\} \\ &= \bigcup_{1 \le i \le n} \{x \in U \mid x \models S_i\} \\ &= \bigcup_{1 \le i \le n} m(\{S_i\}) \\ &= \bigcup_{1 \le i \le n} \{x \in U \mid \bigwedge_{(a,v) \in S_i} x \models (a, v)\} \\ &= \bigcup_{1 \le i \le n} \bigcap_{(a,v) \in S_i} \{x \in U \mid x \models (a, v)\}. \end{aligned} \tag{A.7}$$

□

A.4 Proof of Theorem 4

Theorem 4 DEF(U) *is a Boolean algebra. That is,* DEF(U) *includes the empty set* $\emptyset$ *and the universe* U*, and it is closed under set complement* ^{c}*, intersection* $\cap$ *and union* $\cup$*.*

Proof. This theorem can be proven by verifying the following four facts:

(1) $\emptyset \in \mathrm{DEF}(U)$.

The empty set $\emptyset$ can be the meaning set of any formula that contains a contradiction. For example,

$$\emptyset \quad = \quad m\big((a = v_1) \wedge (a = v_2)\big), \tag{A.8}$$

where $a \in AT$, $v_1, v_2 \in V_a$ and $v_1 \neq v_2$. Equation (A.8) shows a simplest case of a formula indicating an empty set. Since any object can take exactly one value on a specific attribute, any formula that requires different values on one attribute defines the empty set.

(2) $U \in \mathrm{DEF}(U)$.

The universe U can be the meaning set of any formula that covers all possible values of a particular attribute. For example,

$$U \quad = \quad m\big(\bigvee_{v \in V_a} (a = v)\big), \tag{A.9}$$

where $a \in AT$. In the case that more than one attribute appears in a formula, if the formula covers all possible combination values of these attributes, it can also define the universe U.

(3) $\mathrm{DEF}(U)$ is closed under set union $\cup$ and set intersection $\cap$.

Given two definable sets $X = m(p) \in \mathrm{DEF}(U)$ and $Y = m(q) \in \mathrm{DEF}(U)$, according to equations in (2.4), it follows that:

$$\begin{aligned} X \cup Y &= m(p) \cup m(q) &= m(p \vee q), \\ X \cap Y &= m(p) \cap m(q) &= m(p \wedge q). \end{aligned} \tag{A.10}$$

Thus, $X \cup Y, X \cap Y \in \mathrm{DEF}(U)$, that is, intersection and union of two definable sets are also definable. Therefore, DEF(U) is closed under set intersection and union.

(4) DEF(U) is closed under set complement ^{c}.

Suppose $X = m(p) \in$ DEF(U), we need to show its complement $X^c \in$ DEF(U), that is, X^c is also definable. First, we introduce the logic negation $\neg$ to a logic formula. For a formula $p \in DL$, if $X = m(p)$, then $X^c = m(\neg p)$. For an atomic formula, we have:

$$\neg(a = v) \quad = \quad \bigvee_{v' \in V_a, v' \neq v} a = v'. \tag{A.11}$$

If X is defined by an atomic formula $a = v$, then X^c is defined by $\neg(a = v)$, that is,

$$X^c \quad = \quad m(\bigvee_{v' \in V_a, v' \neq v} a = v'). \tag{A.12}$$

X^c is a definable set, since $\bigvee_{v' \in V_a, v' \neq v} a = v'$ is a formula in the description language DL.

Now assume that X is defined by a formula r that may be a non-atomic formula in DL and $X^c = m(\neg r)$. We want to show that $\neg r$ can be expressed as a formula in DL by equivalently eliminating the negation connective. Generally, for any formula r, $\neg r$ can be transformed into a form in which negation connective $\neg$ only applies to atomic formulas by successively apply the De Morgan's laws as follows:

$$\begin{aligned} (1) \qquad \neg(p \wedge q) &= \neg p \vee \neg q, \\ (2) \qquad \neg(p \vee q) &= \neg p \wedge \neg q. \end{aligned} \tag{A.13}$$

Then, by replacing the negation of an atomic formula $\neg(a = v)$ by $\bigvee_{v' \in V_a, v' \neq v} a = v'$, r can be equivalently expressed as a formula in DL, which implies that X^c is definable. Thus, DEF(U) is closed under set complement. $\square$

A.5 Proof of Theorem 5

Theorem 5 *Suppose the adaptive approximations are constructed by using* $Seq_n = \{A_1, A_2, \cdots, A_n\}$*, the following properties hold:*

$$\begin{aligned} (1) \qquad \bigcup \underline{dapr}_n(X) &= \bigcup \underline{bapr}_{E_{A_n}}(X) = \underline{apr}_{E_{A_n}}(X), \\ (2) \qquad \bigcup \overline{dapr}_n(X) &= \bigcup \overline{bapr}_{E_{A_n}}(X) = \overline{apr}_{E_{A_n}}(X). \end{aligned} \tag{A.14}$$

Proof.

According to Theorem 1 which states that the unstructured and structured Pawlak approximations cover the same sets of objects, we only need to prove that:

$$\begin{aligned} \bigcup \underline{dapr}_n(X) &= \underline{apr}_{E_{A_n}}(X), \\ \bigcup \overline{dapr}_n(X) &= \overline{apr}_{E_{A_n}}(X). \end{aligned} \tag{A.15}$$

We use the mathematical induction method to prove it by verifying the following two facts:

(1) The equations hold for equivalence relation $A_0 = \emptyset$, that is:

$$\begin{aligned} \bigcup \underline{dapr}_0(X) &= \underline{apr}_{E_\emptyset}(X), \\ \bigcup \overline{dapr}_0(X) &= \overline{apr}_{E_\emptyset}(X). \end{aligned} \tag{A.16}$$

(2) If the equations hold for i $(0 \leq i \leq n-1)$, that is:

$$\begin{aligned} \bigcup \underline{dapr}_i(X) &= \underline{apr}_{E_{A_i}}(X), \\ \bigcup \overline{dapr}_i(X) &= \overline{apr}_{E_{A_i}}(X), \end{aligned} \tag{A.17}$$

then they hold for $i+1$, that is:

$$\begin{aligned}\bigcup \underline{dapr}_{i+1}(X) &= \underline{apr}_{E_{A_{i+1}}}(X),\\ \bigcup \overline{dapr}_{i+1}(X) &= \overline{apr}_{E_{A_{i+1}}}(X).\end{aligned} \tag{A.18}$$

Fact (1) can be directly verified by the initialization step to construct the adaptive approximations. We give the proof of Fact (2) as follows.

(i) Prove that $\bigcup \underline{dapr}_{i+1}(X) = \underline{apr}_{E_{A_{i+1}}}(X)$.

According to the construction steps of the adaptive approximations, we have:

$$\begin{aligned}\bigcup \underline{dapr}_{i+1}(X) &= \big(\bigcup\{K \in X_{i+1}/E_{A_{i+1}} \mid K \subseteq X\}\big) \;\cup\; \big(\bigcup \underline{dapr}_{i}(X)\big)\\ &= \big(\bigcup\{K \in \big(\bigcup \overline{dapr}_{i}(X) - \bigcup \underline{dapr}_{i}(X)\big)/E_{A_{i+1}} \mid K \subseteq X\}\big)\\ &\cup \;\big(\bigcup \underline{dapr}_{i}(X)\big)\\ &= \big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_i}}(X)\big)/E_{A_{i+1}} \mid K \subseteq X\}\big)\\ &\cup \;\underline{apr}_{E_{A_i}}(X).\end{aligned} \tag{A.19}$$

According to the definition of the structured Pawlak approximations (i.e., Definition 3) and Theorem 1, it follows that:

$$\begin{aligned}\underline{apr}_{E_{A_{i+1}}}(X) &= \bigcup \underline{bapr}_{E_{A_{i+1}}}(X) \;=\; \bigcup\{K \in U/E_{A_{i+1}} \mid K \subseteq X\}\\ &= \big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_i}}(X)\big)/E_{A_{i+1}} \mid K \subseteq X\}\big)\\ &\cup \;\big(\bigcup\{K \in \underline{apr}_{E_{A_i}}(X)/E_{A_{i+1}} \mid K \subseteq X\}\big)\\ &\cup \;\big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X)\big)^c/E_{A_{i+1}} \mid K \subseteq X\}\big).\end{aligned} \tag{A.20}$$

The property (P1) of rough set approximations states that $\underline{apr}_{E_{A_i}}(X) \subseteq X$.

Thus, we have:

$$\forall\, K \in \underline{apr}_{E_{A_i}}(X)/E_{A_{i+1}},\ K \subseteq \underline{apr}_{E_{A_i}}(X) \subseteq X. \tag{A.21}$$

Therefore,

$$\bigcup\{K \in \underline{apr}_{E_{A_i}}(X)/E_{A_{i+1}} \mid K \subseteq X\} \ = \ \underline{apr}_{E_{A_i}}(X). \tag{A.22}$$

The property (P1) also states that $X \subseteq \overline{apr}_{E_{A_i}}(X)$. Therefore, $X \cap \big(\overline{apr}_{E_{A_i}}(X)\big)^c = \emptyset$, which means that:

$$\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X)\big)^c/E_{A_{i+1}} \mid K \subseteq X\} = \emptyset. \tag{A.23}$$

From Equations (A.20), (A.22) and (A.23), it follows that:

$$\begin{aligned}\underline{apr}_{E_{A_{i+1}}}(X) \ &= \ \Big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_i}}(X)\big)/E_{A_{i+1}} \mid K \subseteq X\}\Big) \\ &\cup \ \underline{apr}_{E_{A_i}}(X).\end{aligned} \tag{A.24}$$

By comparing Equations (A.19) and (A.24), we can conclude that:

$$\bigcup \underline{dapr}_{i+1} = \underline{apr}_{E_{A_{i+1}}}(X). \tag{A.25}$$

(ii) Prove that $\bigcup \overline{dapr}_{i+1}(X) = \overline{apr}_{E_{A_{i+1}}}(X)$.

According to the construction steps of the adaptive approximations, we have:

$$
\begin{aligned}
\bigcup \overline{dapr}_{i+1}(X) &= \Big(\bigcup\{K \in X_{i+1}/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\Big) \cup \Big(\bigcup \underline{dapr}_{i+1}(X)\Big) \\
&= \Big(\bigcup\{K \in \big(\bigcup \overline{dapr}_i(X) - \bigcup \underline{dapr}_i(X)\big)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\Big) \\
&\cup \Big(\bigcup \underline{dapr}_{i+1}(X)\Big) \\
&= \Big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_i}}(X)\big)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\Big) \\
&\cup \underline{apr}_{E_{A_{i+1}}}(X) \\
&= \Big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_{i+1}}}(X)\big)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\} \\
&\cup \Big(\bigcup\{K \in \big(\underline{apr}_{E_{A_{i+1}}}(X) - \underline{apr}_{E_{A_i}}(X)\big)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\} \\
&\cup \underline{apr}_{E_{A_{i+1}}}(X). \qquad \text{(A.26)}
\end{aligned}
$$

By noticing that:

$$
\bigcup\{K \in \big(\underline{apr}_{E_{A_{i+1}}}(X) - \underline{apr}_{E_{A_i}}(X)\big)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\} \subseteq \underline{apr}_{E_{A_{i+1}}}(X), \qquad \text{(A.27)}
$$

we can simplify Equation (A.26) as:

$$
\begin{aligned}
\bigcup \overline{dapr}_{i+1}(X) &= \Big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_{i+1}}}(X)\big)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\Big) \\
&\cup \underline{apr}_{E_{A_{i+1}}}(X). \qquad \text{(A.28)}
\end{aligned}
$$

According to the definition of the structured Pawlak approximations (i.e., Definition 3) and Theorem 1, it follows that:

$$
\begin{aligned}
\overline{apr}_{E_{A_{i+1}}}(X) &= \bigcup \underline{bapr}_{E_{A_{i+1}}}(X) = \bigcup\{K \in U/E_{A_{i+1}} \mid K \cap X \neq \emptyset\} \\
&= \Big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_{i+1}}}(X)\big)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\Big) \\
&\cup \Big(\bigcup\{K \in \underline{apr}_{E_{A_{i+1}}}(X)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\Big) \\
&\cup \Big(\bigcup\{K \in \big(\overline{apr}_{E_{A_i}}(X)\big)^c/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\Big). \qquad \text{(A.29)}
\end{aligned}
$$

From the property (P1) of rough set approximations, we have:

$$\forall\, K \in \underline{apr}_{E_{A_{i+1}}}(X)/E_{A_{i+1}},\ K \subseteq \underline{apr}_{E_{A_{i+1}}}(X) \subseteq X, \tag{A.30}$$

which means that $K \cap X = K \neq \emptyset$. Therefore,

$$\bigcup\{K \in \underline{apr}_{E_{A_{i+1}}}(X)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\} \;=\; \underline{apr}_{E_{A_{i+1}}}(X). \tag{A.31}$$

The property (P1) states that $X \subseteq \overline{apr}_{E_{A_i}}(X)$ which means that $X \cap \left(\overline{apr}_{E_{A_i}}(X)\right)^c = \emptyset$. Thus,

$$\bigcup\{K \in \left(\overline{apr}_{E_{A_i}}(X)\right)^c/E_{A_{i+1}} \mid K \cap X \neq \emptyset\} = \emptyset. \tag{A.32}$$

From Equations (A.29), (A.31) and (A.32), it follows that:

$$\begin{aligned}\overline{apr}_{E_{A_{i+1}}}(X) \;&=\; \left(\bigcup\{K \in \left(\overline{apr}_{E_{A_i}}(X) - \underline{apr}_{E_{A_{i+1}}}(X)\right)/E_{A_{i+1}} \mid K \cap X \neq \emptyset\}\right)\\ &\cup\; \underline{apr}_{E_{A_{i+1}}}(X).\end{aligned} \tag{A.33}$$

By comparing Equations (A.28) and (A.33), we can conclude that:

$$\bigcup \overline{dapr}_{i+1} = \overline{apr}_{E_{A_{i+1}}}(X). \tag{A.34}$$

$\square$

A.6 Proof of Theorem 6

Theorem 6 *The following properties hold:*

$$\begin{aligned}&(1) \qquad \forall\, A \subseteq AT,\ \underline{bapr}_{E_A}(X) \subseteq \underline{sapr}(X),\\ &(2) \qquad \underline{apr}_{E_{AT}}(X) = \bigcup \underline{sapr}(X).\end{aligned} \tag{A.35}$$

Proof.

(1) Prove that $\underline{bapr}_{E_A}(X) \subseteq \underline{sapr}(X)$.

Each equivalence class $[x]_{E_A}$ is a nonempty conjunctively definable set since it can be expressed as:

$$[x]_{E_A} \quad = \quad m\Big(\bigwedge_{a \in A} (a = I_a(x))\Big). \tag{A.36}$$

By Definition 3, each equivalence class $[x]_{E_A} \in \underline{bapr}_{E_A}(X)$ is a nonempty conjunctively definable subset of X. By Definition 16, it follows that $[x]_{E_A} \in \underline{sapr}(X)$. Therefore, it can be concluded that $\underline{bapr}_{E_A} \subseteq \underline{sapr}(X)$.

(2) Prove that $\underline{apr}_{E_{AT}}(X) = \bigcup \underline{sapr}(X)$.

According to set theory, it can be proved by verifying the following two subset relationships:

(i) $\underline{apr}_{E_{AT}}(X) \subseteq \bigcup \underline{sapr}(X)$:

The first property in this theorem tells that $\underline{bapr}_{E_{AT}}(X) \subseteq \underline{sapr}(X)$. Thus, the sets of objects covered by these two approximations have the similar subset relationship, that is, $\bigcup \underline{bapr}_{E_{AT}}(X) \subseteq \bigcup \underline{sapr}(X)$. According to Theorem 1, $\underline{apr}_{E_{AT}}(X) = \bigcup \underline{bapr}_{E_{AT}}(X)$. So we can conclude that $\underline{apr}_{E_{AT}}(X) \subseteq \bigcup \underline{sapr}(X)$.

(ii) $\underline{apr}_{E_{AT}}(X) \supseteq \bigcup \underline{sapr}(X)$:

According to the definition of the structured lower approximations (i.e., Definition 16), for any object x which is covered by $\underline{sapr}(X)$, there must exist a nonempty conjunctively definable set C such that $x \in C$ and $C \subseteq X$. In Section 2.3.3, we verified the property that a conjunctively definable set can be expressed as the union of a family of definable elementary sets, that is, the equivalence classes in U/E_{AT}. Therefore, C can be expressed by the union of a family $\mathbb{F} \subseteq U/E_{AT}$, that is, $C = \bigcup \mathbb{F}$. And there must exist an equivalence class $[y]_{E_{AT}}$ in the family such that $x \in [y]_{E_{AT}}$ and $[y]_{E_{AT}} \subseteq C$. Thus, $[x]_{E_{AT}} = [y]_{E_{AT}} \subseteq C \subseteq X$. This means that $x \in \underline{apr}_{E_{AT}}(X)$. Then it can be concluded that $\underline{apr}_{E_{AT}}(X) \supseteq \bigcup \underline{sapr}(X)$.

$\square$

A.7 Proof of Theorem 7

Theorem 7 *The following properties hold:*

$$
\begin{array}{ll}
(1) & \text{For any sequence } Seq_n \text{ of sets of attributes, } \underline{dapr}_n(X) \subseteq \underline{sapr}(X), \\
(2) & \text{For } Seq_{|AT|} = \{A_1, A_2, \cdots, AT\},\ \bigcup \underline{dapr}_{|AT|}(X) = \bigcup \underline{sapr}(X).
\end{array}
\tag{A.37}
$$

Proof.

(1) Prove that for any sequence Seq_n of sets of attributes, $\underline{dapr}_n(X) \subseteq \underline{sapr}(X)$.

In the proof of Theorem 6, we verified that each equivalence class $[x]_{E_A}$ $(A \subseteq AT)$ is a nonempty conjunctively definable set. From the construction of the adaptive approximations, it follows that each equivalence class in $\underline{dapr}_n(X)$ is a nonempty conjunctively definable subset of X, which means that it is in $\underline{sapr}(X)$. Therefore, it can be concluded that $\underline{dapr}_n(X) \subseteq \underline{sapr}(X)$.

(2) Prove that $\bigcup \underline{dapr}_{|AT|}(X) = \bigcup \underline{sapr}(X)$.

From Theorem 5, it follows that $\bigcup \underline{dapr}_{|AT|}(X) = \underline{apr}_{E_{AT}}(X)$. Theorem 6 states that $\underline{apr}_{E_{AT}}(X) = \bigcup \underline{sapr}(X)$. Therefore, it can be concluded that $\bigcup \underline{dapr}_{|AT|}(X) = \bigcup \underline{sapr}(X)$.

$\square$

A.8 Proof of Theorem 8

Theorem 8 *The following properties hold for $R_{\cup}(sapr(X))$:*

$$
\begin{aligned}
&(1) \quad \forall \mathbb{R} \in R_{\cup}(sapr(X)),\\
&\qquad \bigcup \mathbb{R} = \bigcup \underline{sapr}(X) = \underline{apr}_{E_{AT}}(X) = \bigcup \underline{bapr}_{E_{AT}}(X) = \bigcup \underline{dapr}_{|AT|}(X),\\
&(2) \quad \underline{bapr}_{E_{AT}}(X) \in R_{\cup}(sapr(X)),\\
&(3) \quad \underline{dapr}_{|AT|}(X) \in R_{\cup}(sapr(X)).
\end{aligned}
\tag{A.38}
$$

Proof.

(1) Prove that $\bigcup \mathbb{R} = \bigcup \underline{sapr}(X) = \underline{apr}_{E_{AT}}(X) = \bigcup \underline{bapr}_{E_{AT}}(X) = \bigcup \underline{dapr}_{|AT|}(X)$.

These equal signs can be directly verified by Definition 17, Theorem 6, Theorem 1 and Theorem 5, respectively.

(2) Prove that $\underline{bapr}_{E_{AT}}(X) \in R_{\cup}(sapr(X))$.

According to the definition of a $\cup$-reduct, we need to prove the following two facts:

(i) $\bigcup \underline{bapr}_{E_{AT}}(X) = \bigcup \underline{sapr}(X)$:

This can be directly proved by Theorem 6 which states that $\underline{apr}_{E_{AT}}(X) = \bigcup \underline{sapr}(X)$ and Theorem 1 which states that $\underline{apr}_{E_{AT}}(X) = \bigcup \underline{bapr}_{E_{AT}}(X)$.

(ii) $\forall [x]_{E_{AT}} \in \underline{bapr}_{E_{AT}}(X), \bigcup(\underline{bapr}_{E_{AT}}(X) - \{[x]_{E_{AT}}\}) \neq \bigcup \underline{sapr}(X)$:

Since Bryniarski approximations are partition-based, removing any equivalence class in $\underline{bapr}_{E_{AT}}(X)$ will change the objects covered. That is, $\forall [x]_{E_{AT}} \in \underline{bapr}_{E_{AT}}(X), \bigcup(\underline{bapr}_{E_{AT}}(X) - \{[x]_{E_{AT}}\}) \neq \bigcup \underline{bapr}_{E_{AT}}(X)$. We have proved that $\bigcup \underline{bapr}_{E_{AT}}(X) = \bigcup \underline{sapr}(X)$, so $\forall [x]_{E_{AT}} \in \underline{bapr}_{E_{AT}}(X), \bigcup(\underline{bapr}_{E_{AT}}(X) - \{[x]_{E_{AT}}\}) \neq \bigcup \underline{sapr}(X)$.

(3) Prove that $\underline{dapr}_{|AT|}(X) \in R_{\cup}(sapr(X))$.

According to the definition of a $\cup$-reduct, we need to prove the following two facts:

(i) $\bigcup \underline{dapr}_{|AT|}(X) = \bigcup \underline{sapr}(X)$:

This equation is given in Theorem 7.

(ii) $\forall K \in \underline{dapr}_{|AT|}(X), \bigcup(\underline{dapr}_{|AT|}(X) - \{K\}) \neq \bigcup \underline{sapr}(X)$:

Since the adaptive approximations are partition-based, removing any set from $\underline{dapr}_{|AT|}(X)$ will change the set of objects covered. That is, $\forall K \in \underline{dapr}_{|AT|}(X), \bigcup(\underline{dapr}_{|AT|}(X) - \{K\}) \neq \bigcup \underline{dapr}_{|AT|}(X)$. We have proved that $\bigcup \underline{dapr}_{|AT|}(X) = \bigcup \underline{sapr}(X)$. Thus, $\forall K \in \underline{dapr}_{|AT|}(X), \bigcup(\underline{dapr}_{|AT|}(X) - \{K\}) \neq \bigcup \underline{sapr}(X)$.

$\square$

A.9 Proof of Theorem 9

Theorem 9 *The following properties hold:*

$$\begin{aligned} (1) \qquad & \mathbb{M}_{ex}(\underline{sapr}(X)) \subseteq \underline{sapr}(X), \\ (2) \qquad & \bigcup \mathbb{M}_{ex}(\underline{sapr}(X)) = \bigcup \underline{sapr}(X). \end{aligned} \tag{A.39}$$

That is, $\mathbb{M}_{ex}(\underline{sapr}(X))$ *is a subset of* $\underline{sapr}(X)$*, which covers the same set of objects as* $\underline{sapr}(X)$*.*

Proof.

The first equation can be derived directly from the definition of extension-based maximal elements, that is, Definition 19.

The proof of the second equation is given as follows. From the first equation, it follows that:

$$\bigcup \mathbb{M}_{ex}(\underline{sapr}(X)) \;\subseteq\; \bigcup \underline{sapr}(X). \tag{A.40}$$

Suppose $\bigcup \mathbb{M}_{ex}(\underline{sapr}(X)) \neq \bigcup \underline{sapr}(X)$. There exists an object x_0 such that $x_0 \in \bigcup \underline{sapr}(X)$ and $x_0 \notin \bigcup \mathbb{M}_{ex}(\underline{sapr}(X))$. Suppose a conjunctively definable set $C \in \underline{sapr}(X)$ contains x_0, that is, $x_0 \in C$. Since $x_0 \notin \bigcup \mathbb{M}_{ex}(\underline{sapr}(X))$, it follows that $C \notin \mathbb{M}_{ex}(\underline{sapr}(X))$, which means there exists an extension-based maximal element $C_M \in \underline{sapr}(X)$ such that $C \subsetneq C_M$. Therefore, $x_0 \in C_M$. Then it follows that $x_0 \in \bigcup \mathbb{M}_{ex}(\underline{sapr}(X))$, which produces a contradiction. Thus, it can be concluded that $\bigcup \mathbb{M}_{ex}(\underline{sapr}(X)) = \bigcup \underline{sapr}(X)$.

□

A.10 Proof of Theorem 10

Theorem 10 *Let $R_\cup(\cdot)$ denote the family of all $\cup$-reducts. The following property holds:*

$$R_\cup(\mathbb{M}_{ex}(\underline{sapr}(X))) \;\subseteq\; R_\cup(\underline{sapr}(X)). \tag{A.41}$$

Proof.

For any $\mathbb{R} \in R_\cup(\mathbb{M}_{ex}(\underline{sapr}(X)))$, $\mathbb{R}$ is a $\cup$-reduct of $\underline{sapr}(X)$ with maximal elements. This means that $\mathbb{R}$ is a $\cup$-reduct of $\underline{sapr}(X)$, that is, $\mathbb{R} \in R_\cup(\underline{sapr}(X))$. Therefore, it can be concluded that $R_\cup(\mathbb{M}_{ex}(\underline{sapr}(X))) \subseteq R_\cup(\underline{sapr}(X))$.

□

REFERENCES

[1] F. Baader. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[2] Z. Bonikowski, E. Bryniarski, and U. Wybraniec-Skardowska. Extensions and intensions in the rough set theory. *Information Sciences*, 107:149–167, 1998.

[3] E. Bryniarski. A calculus of rough sets of the first order. *Bulletin of the Polish Academy of Sciences*, 37:71–78, 1989.

[4] W. Buszkowski. Approximation spaces and definability for incomplete information systems. In L. Polkowski and A. Skowron, editors, *Rough Sets and Current Trends in Computing*, volume 1424 of *LNCS*, pages 115–122. Springer, Heidelberg, 1998.

[5] S. Carey. *The Origin of Concepts*. Oxford University Press, 2009.

[6] X.F. Deng, H.K. Wang, and Y.Y. Yao. Adaptive rough sets approximations based on three-way decisions (in Chinese). In X.Y. Jia, L. Shang, X.Z. Zhou, J.Y. Liang, D.Q. Miao, G.Y. Wang, T.R. Li, and Y.P. Zhang, editors, *Theory of Three-way Decisions and Application*, pages 118–130. Nanjing University Press, Nanjing, China, 2012.

[7] D. Dubois and H. Prade. Rough fuzzy sets and fuzzy rough sets. *International Journal of General System*, 17:191–209, 1990.

[8] U. Eco. *A Theory of Semiotics*, volume 217. Indiana University Press, 1976.

[9] J.W. Grzymała-Busse. LERS - A system for learning from examples based on rough sets. In R. Słowiński, editor, *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, pages 3–18. Kluwer Academic Publishers, Dordrecht, 1992.

[10] J.W. Grzymała-Busse. A new version of the rule induction system LERS. *Fundamenta Informaticae*, 31:27–39, 1997.

[11] M. Hazewinkel. *Encyclopedia of Mathematics*, volume 9. Kluwer Academic Publishers, Dordrecht, 1993.

[12] T.P. Hong, L.H. Tseng, and S.L. Wang. Learning rules from incomplete training examples by rough sets. *Expert Systems with Applications*, 22:285–293, 2002.

[13] M. Kryszkiewicz. Rules in incomplete information systems. *Information Sciences*, 113:271–292, 1999.

[14] R. Li and Y.Y. Yao. Indiscernibility and similarity in an incomplete information table. In J. Yu, S. Greco, P. Lingras, G. Wang, and A. Skowron, editors, *Rough Set and Knowledge Technology*, volume 6401 of *LNCS*, pages 110–117. Springer, Heidelberg, 2010.

[15] W.J. Lipski. On semantics issues connected with incomplete information databases. *ACM Transactions on Database Systems*, 4:262–296, 1979.

[16] W.J. Lipski. On databases with incomplete information. *Journal of the Association of Computing Machinery*, 28:41–70, 1981.

[17] W. Marek and Z. Pawlak. Information storage and retrieval systems: Mathematical foundations. *Theoretical Computer Science*, 1:331–354, 1976.

[18] E. Margolis and S. Laurence. Concepts. In E.N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. 2012.

[19] G.D. Martin. *Language, Truth and Poetry: Notes Towards a Philosophy of Literature*. Edinburgh University Press, 1975.

[20] R.S. Michalski, J.G. Carbonell, and T.M. Mitchell. *Machine Learning, an Artificial Intelligence Approach*. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1983.

[21] G. Murphy. *The Big Book of Concepts*. The MIT Press, 2002.

[22] C.K. Ogden and I.A. Richards. *The Meaning of Meaning*. Harcourt, Brace, 1923.

[23] E. Orlowska. *Incomplete Information: Rough Set Analysis*, volume 13. Springer, Heidelberg, 1998.

[24] Z. Pawlak. Rough sets. *International Journal of Computer and Information Sciences*, 11:341–356, 1982.

[25] Z. Pawlak. Rough classification. *International Journal of Man-Machine Studies*, 20:469–483, 1984.

[26] Z. Pawlak. *Rough Sets, Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Dordrecht, 1991.

[27] J.R. Quinlan. Learning efficient classification procedures and their application to chess end games. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning*, Symbolic Computation, pages 463–482. Springer, Heidelberg, 1983.

[28] S. Regoczei and G. Hirst. Knowledge acquisition as knowledge explication by conceptual analysis. In J.F. Sowa, N. Foo, and A. Rao, editors, *Conceptual Graphs for Knowledge Systems*. 1990.

[29] M. Restrepo, C. Cornelis, and J. Gómez. Duality, conjugacy and adjointness of approximation operators in covering-based rough sets. *International Journal of Approximate Reasoning*, 55:469–485, 2014.

[30] R. Słowiński and J. Stefanowski. Rough-set reasoning about uncertain data. *Fundamenta Informaticae*, 27:229–243, 1996.

[31] E.E. Smith. Concepts and induction. In M.I. Posner, editor, *Foundations of Cognitive Science*, pages 501–526. The MIT Press, Cambridge, Massachusetts, 1989.

[32] J.F. Sowa. *Conceptual Structures, Information Processing in Mind and Machine.* Addison-Wesley, Reading, Massachusetts, 1984.

[33] J. Stefanowski and A. Tsoukiàs. On the extension of rough sets under incomplete information. In N. Zhong, A. Skowron, and S. Ohsuga, editors, *New Directions in Rough Sets, Data Mining and Granular-Soft Computing*, volume 1711 of *LNCS*, pages 73–81. Springer, Heidelberg, 1999.

[34] I. van Mechelen, J. Hampton, R.S. Michalski, and P. Thenus. *Categories and Concepts: Theoretical Views and Inductive Data Analysis.* Academic Press, New York, 1993.

[35] G.Y. Wang. Extension of rough set under incomplete information systems. In *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems*, volume 2, pages 1098–1103, 2002.

[36] Y.Y. Yao. On generalizing rough set theory. In G.Y. Wang, Q. Liu, Y.Y. Yao, and A. Skowron, editors, *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, volume 2639 of *LNCS*, pages 44–51. Springer, Heidelberg, 2003.

[37] Y.Y. Yao. A partition model of granular computing. In J.F. Peters, A. Skowron, J.W. Grzymała-Busse, B. Kostek, R.W. Świniarski, and M.S. Szczuka, editors, *Transactions on Rough Sets I*, volume 3100 of *LNCS*, pages 232–253. Springer, Heidelberg, 2004.

[38] Y.Y. Yao. A note on definability and approximations. In J.F. Peters, A. Skowron, V.W. Marek, E. Orłowska, R. Słowiński, and W. Ziarko, editors, *Transactions on Rough Sets VII*, volume 4400 of *LNCS*, pages 274–282. Springer, Heidelberg, 2007.

[39] Y.Y. Yao. Interpreting concept learning in cognitive informatics and granular computing. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39:855–866, Aug 2009.

[40] Y.Y. Yao. Interval sets (in chinese). In G.Y. Wang, D.Y. Li, Y.Y. Yao, J.Y. Liang, D.Q. Miao, Y.P. Zhang, and Q.H. Zhang, editors, *Cloud Model and Granular Computing*, pages 74–93. Science Press, Beiijing, 2012.

[41] Y.Y. Yao. Duality in rough set theory based on the square of opposition. *Fundamenta Informaticae*, 127:49–64, 2013.

[42] Y.Y. Yao. Rough set approximations: A concept analysis point of view. Working paper, 2013.

[43] Y.Y. Yao and X.F. Deng. A granular computing paradigm for concept learning. In S. Ramanna, L. Jain, and R.J. Howlett, editors, *Emerging Paradigms in Machine Learning*, volume 13, pages 307–326. Springer, Heidelberg, 2013.

[44] Y.Y. Yao and R. Fu. The concept of reducts in Pawlak three-step rough set analysis. In J.F. Peters, A. Skowron, S. Ramanna, Z. Suraj, and X. Wang, editors, *Transactions on Rough Sets XVI*, volume 7736 of *LNCS*, pages 53–72. Springer, Heidelberg, 2013.

[45] Y.Y. Yao and M.J. Hu. A definition of structured rough set approximations. In M. Kryszkiewicz, C. Cornelis, D. Ciucci, J. Medina-Moreno, H. Motoda, and Z.W. Raś, editors, *Rough Sets and Intelligent Systems Paradigms*, volume 8537 of *LNCS*, pages 111–122. Springer International Publishing, 2014.

[46] Y.Y. Yao and M.J. Hu. Interval sets: Analysis of incomplete information (manuscript). 2014.

[47] Y.Y. Yao and N. Noroozi. A unified framework for set-based computations. In T.Y. Lin, editor, *Proceedings of the 3rd International Workshop on Rough Sets and Soft Computing*, pages 236–243, San Jose, California, USA, November 10-12 1994. San Jose State University.

[48] Y.Y. Yao and B.X. Yao. Covering based rough set approximations. *Information Sciences*, 200:91–107, 2012.

[49] Y.Y. Yao and Y. Zhao. Attribute reduction in decision-theoretic rough set models. *Information Sciences*, 178:3356–3373, 2008.

[50] Y.Y. Yao and Y. Zhao. Discernibility matrix simplification for constructing attribute reducts. *Information Sciences*, 179:867–882, 2009.

[51] Y.Y. Yao, Y. Zhao, and J. Wang. On reduct construction algorithms. In G.Y. Wang, J.F. Peters, A. Skowron, and Y.Y. Yao, editors, *Rough Sets and Knowledge Technology*, volume 4062 of *LNCS*, pages 297–304. Springer, Heidelberg, 2006.

[52] W.F. Żakowski. Approximations in the space (U, Π). *Demonstratio Mathematica*, 16:761–769, 1983.

[53] Y. Zhao, F. Luo, S.K.M. Wong, and Y.Y. Yao. A general definition of an attribute reduct. In J.T. Yao, P. Lingras, W.Z. Wu, M. Szczuka, N.J. Cercone,

and D. Slezak, editors, *Rough Sets and Knowledge Technology*, volume 4481 of *LNCS*, pages 101–108. Springer, Heidelberg, 2007.

[54] W. Zhu and F.Y. Wang. Reduction and axiomization of covering generalized rough sets. *Information Sciences*, 152:217–230, 2003.

[55] W. Zhu and F.Y. Wang. On three types of covering-based rough sets. *IEEE Transctions on Knowledge and Data Engineering*, 19:1131–1144, August 2007.