Concept Lattices in Rough Set Theory

Y.Y. Yao

Department of Computer Science, University of Regina
Regina, Saskatchewan, Canada S4S 0A2
E-mail: yyao@cs.uregina.ca
URL: http://www.cs.uregina/∼yyao

***Abstract*— An alternative formulation of rough set theory can be developed based on a binary relation between two universes, one is a finite set of objects and the other is a finite set of properties. Rough set approximation operators are defined with respect to the binary relation. Three concept lattices are constructed based on approximation operators. They are different from, but related to, the concept lattice built in formal concept analysis. Through the study of the introduced concept lattices, one can obtain an in-depth understanding of data analysis using rough set theory.**

TABLE I
A FORMAL CONTEXT

	a	b	c	d	e
1	×		×	×	×
2	×		×		
3		×			×
4		×			×
5	×				
6	×	×			×

I. INTRODUCTION

Rough set theory [7], [8] and formal concept analysis [2], [9] provide two related methods for data analysis [3], [10]. They study and model the notion of concepts from different perspectives. The central notions of rough set theory are the indiscernibility of objects with respect to a set of properties and the induced approximation operators. The central notions of formal concept analysis are formal concepts and concept lattice. Proposals have been made to combine these two theories in a common framework [1], [3], [4], [5], [10], [11]. On the one hand, one can introduce the notion of concept lattice into rough set theory [3], [10]. On the other hand, one can introduce the notion of approximation operators into formal concept analysis [4], [5], [6], [11].

The classical rough set theory is developed based on an equivalence (indiscernibility) relation on a universe of objects. Generalized formulation has been proposed by using a binary relation on two universes, one is a set of objects and the other is a set of properties [3], [13]. A binary relation on two universes is commonly known as a formal context in formal concept analysis. It serves as a common basis for rough set theory and formal concept analysis.

Düntsch and Gediga, following the study of modal logics, defined modal-style operators based on a binary relation [1], [3]. The derivation operator of formal concept analysis is a polarity or sufficiency operator used in modal logics, and the rough set approximation operators are in fact the necessity and possibility operators used in modal logics [3], [12]. The formulation in terms of modal-style operators enables us to gain more insights into theories of rough sets and formal concept analysis.

Düntsch and Gediga introduced a concept lattice constructed based on approximation operators [3]. Yao introduced another concept lattice and compared the roles of different concept lattices in data analysis [10]. The objective of this paper is to extend the results of these studies and to provide a more systematic examination of concept lattices in rough set theory.

II. FORMAL CONTEXTS AND APPROXIMATION OPERATORS

Let U and V be any two finite nonempty sets. Elements of U are called objects, and elements of V are called properties or attributes. The set of objects U is described by the set of properties. More precisely, the relationships between objects and properties are described by a binary relation R between U and V, which is a subset of the Cartesian product $U \times V$. The triplet (U, V, R) is called a formal context in formal concept analysis, and an approximation space in rough set theory.

In a formal context (U, V, R), for a pair of elements $x \in U$ and $y \in V$, if $(x, y) \in R$, also written as xRy, we say that x has the property y, or the property y is possessed by object x. Table I is an example of a formal context. In this table, for example, object 1 has properties a, c, d, and e. The property a is possessed by objects $1, 2, 5$, and 6.

The binary relation can be equivalently expressed in two additional forms. An object $x \in U$ has the set of properties:

$$xR = \{y \in V \mid xRy\} \subseteq V. \tag{1}$$

A property y is possessed by the set of objects:

$$Ry = \{x \in U \mid xRy\} \subseteq U. \tag{2}$$

The three different representations uniquely determine each other, namely, $xRy \iff y \in xR \iff x \in Ry$. One can extend xR to a subset $X \subseteq U$ and Ry to a subset $Y \subseteq V$:

$$\begin{aligned} XR &= \bigcup_{x\in X} xR, \\ RY &= \bigcup_{y\in Y} Ry. \end{aligned} \quad (3)$$

A property in XR is possessed by *at least one* object in X, and an object in RY has *at least one* property in Y.

With respect to a formal context (U, V, R), we define a pair of dual approximation operators, ${}^\Box, {}^\Diamond : 2^U \longrightarrow 2^V$:

$$\begin{aligned} X^\Box &= \{y \in V \mid \forall x \in U(xRy \Longrightarrow x \in X)\} \\ &= \{y \in V \mid Ry \subseteq X\}, \quad (4) \\ X^\Diamond &= \{y \in V \mid \exists x \in U(xRy \wedge x \in X)\} \\ &= \{y \in V \mid Ry \cap X \neq \emptyset\} \\ &= \bigcup_{x\in X} xR \\ &= XR. \quad (5) \end{aligned}$$

They are related by $X^{c\Box c} = X^\Diamond$ and $X^{c\Diamond c} = X^\Box$, where ^{c} denotes the complement of a set. By definition, an object having a property in $X^\Box$ is *necessarily* in X, and an object having a property in $X^\Diamond$ is only *possibly* in X. Thus, the operators ${}^\Box$ and ${}^\Diamond$ are also referred to as the necessity and the possibility operators [3].

If we assume that $Ry \neq \emptyset$ for all $y \in V$, namely, each property must be possessed by at least one object in U, we have $X^\Box \subseteq X^\Diamond$.

Conversely, we define a pair of dual approximation operators, ${}^\Box, {}^\Diamond : 2^V \longrightarrow 2^U$:

$$\begin{aligned} Y^\Box &= \{x \in U \mid \forall y \in V(xRy \Longrightarrow y \in Y)\} \\ &= \{x \in U \mid xR \subseteq Y\}, \quad (6) \\ Y^\Diamond &= \{x \in U \mid \exists y \in V(xRy \wedge y \in Y)\} \\ &= \{x \in U \mid xR \cap Y \neq \emptyset\} \\ &= \bigcup_{y\in Y} Ry, \\ &= RY. \quad (7) \end{aligned}$$

The same symbols are used for both operators from 2^U to 2^V, and from 2^V to 2^U. Their roles can be easily seen from the context.

The pair of approximation operators have the properties: for $X, X_1, X_2 \subseteq U$ and $Y, Y_1, Y_2 \subseteq V$,

(i) $X_1 \subseteq X_2 \Longrightarrow [X_1^\Box \subseteq X_2^\Box, X_1^\Diamond \subseteq X_2^\Diamond]$,
$Y_1 \subseteq Y_2 \Longrightarrow [Y_1^\Box \subseteq Y_2^\Box, Y_1^\Diamond \subseteq Y_2^\Diamond]$,

(ii) $X^{\Box\Diamond} \subseteq X \subseteq X^{\Diamond\Box}$,
$Y^{\Box\Diamond} \subseteq Y \subseteq Y^{\Diamond\Box}$,

(iii) $X^{\Diamond\Box\Diamond} = X^\Diamond$,
$X^{\Box\Diamond\Box} = X^\Box$,
$Y^{\Diamond\Box\Diamond} = Y^\Diamond$,
$Y^{\Box\Diamond\Box} = Y^\Box$,

(iv) $(X_1 \cap X_2)^\Box = X_1^\Box \cap X_2^\Box$,
$(X_1 \cup X_2)^\Diamond = X_1^\Diamond \cup X_2^\Diamond$,
$(Y_1 \cap Y_2)^\Box = Y_1^\Box \cap Y_2^\Box$,
$(Y_1 \cup Y_2)^\Diamond = Y_1^\Diamond \cup Y_2^\Diamond$.

As an example, we show that $X^{\Box\Diamond\Box} = X^\Box$.
$(\Longrightarrow)$ Assume $y \in X^{\Box\Diamond\Box}$, we have $Ry \subseteq X^{\Box\Diamond}$. Therefore, for all $x \in U$, $xRy \Longrightarrow x \in X^{\Box\Diamond}$, namely, $xRy \Longrightarrow xR \cap X^\Box \neq \emptyset$. Thus, for all $x \in U$, xRy implies that there exists a $z \in V$ such that $xRz \wedge z \in X^\Box$. The condition $z \in X^\Box$ implies $Rz \subseteq X$. Combining it with xRz results in $x \in X$. Therefore, for all $x \in U$, xRy implies $x \in X$. That is, $y \in X^\Box$.
$(\Longleftarrow)$ Assume that $y \in X^\Box$. Then for all $x \in U$, $xRy \Longrightarrow xR \cap X^\Box \neq \emptyset$. It implies that for all $x \in U$, $xRy \Longrightarrow x \in X^{\Box\Diamond}$. It follows that $y \in X^{\Box\Diamond\Box}$.
The equality $X^{\Diamond\Box\Diamond} = X^\Diamond$ follows from $X^{\Box\Diamond\Box} = X^\Box$ by the duality of ${}^\Box$ and ${}^\Diamond$.

III. Formal Concept Lattices based on Approximation Operators

Based on the notion of approximation operators, three concept lattices are introduced [3], [10].

A. The object oriented concept lattice

The object oriented concept lattice was introduced by Yao [10]. A pair (X, Y), $X \subseteq U, Y \subseteq V$, is called an object oriented formal concept if $X = Y^\Diamond$ and $Y = X^\Box$. If an object has a property in Y then the object belongs to X. Furthermore, only objects in X have properties in Y. The set of objects X is called the extension of the concept (X, Y), and the set of the properties Y is called the intension. The intension can be used to construct the expression $\bigvee_{y\in Y} xRy$ that describes the objects in the extension.

The family of all object oriented concepts forms a lattice. The meet $\wedge$ and the join $\vee$ are defined by:

$$\begin{aligned} (X_1, Y_1) \wedge (X_2, Y_2) &= ((Y_1 \cap Y_2)^\Diamond, Y_1 \cap Y_2) \\ &= ((X_1 \cap X_2)^{\Box\Diamond}, Y_1 \cap Y_2), \\ (X_1, Y_1) \vee (X_2, Y_2) &= (X_1 \cup X_2, (X_1 \cup X_2)^\Box) \\ &= (X_1 \cup X_2, (Y_1 \cup Y_2)^{\Diamond\Box}). \end{aligned}$$

From the definition and properties (iii) and (iv), one can verify that the pair $((Y_1 \cap Y_2)^\Diamond, Y_1 \cap Y_2)$ is an object oriented concept. More specifically,

$$\begin{aligned} ((Y_1 \cap Y_2)^\Diamond)^\Box &= ((X_1^\Box \cap X_2^\Box)^\Diamond)^\Box) \\ &= (X_1 \cap X_2)^{\Box\Diamond\Box} \\ &= (X_1 \cap X_2)^\Box \\ &= X_1^\Box \cap X_2^\Box \\ &= Y_1 \cap Y_2. \end{aligned} \quad (8)$$

The pair $(X_1 \cup X_2, (X_1 \cup X_2)^\Box)$ is also an object oriented concept.

For a set of objects $X \subseteq U$, from property (iii) we have $(X^{\Box\Diamond})^\Box = X^\Box$ and hence an object oriented concept

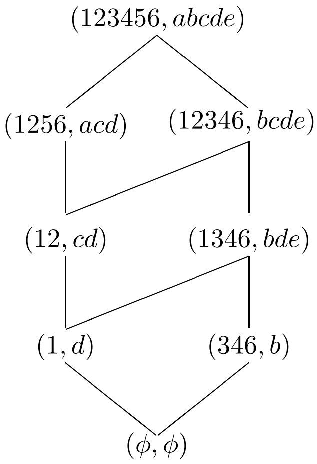

Fig. 1. An object oriented concept lattice

$(X^{\Box\Diamond}, X^{\Box})$. For a set of properties $Y \subseteq V$, we have another object oriented concept $(Y^{\Diamond}, Y^{\Diamond\Box})$. In the special case, for a single attribute $y \in V$, we have an object oriented concept $(\{y\}^{\Diamond}, \{y\}^{\Diamond\Box}) = (Ry, (Ry)^{\Box})$.

For two object oriented concepts (X_1, Y_1) and (X_2, Y_2), we say that (X_1, Y_1) is a sub-concept of (X_2, Y_2) if and only if $X_1 \subseteq X_2$, or equivalently, if and only if $Y_1 \subseteq Y_2$.

For the formal context of Table I, the corresponding object oriented formal concept lattice is given by Figure 1. For simplicity, a set is denoted by listing its elements. For example, the set of objects $\{1, 2, 5, 6\}$ is denoted by 1256.

For a set of objects $X \subseteq U$, we can construct a set of properties $X^{\Box}$. It can be used to derive rules that determine whether an object is in X. If an object has a property in $X^{\Box}$, the object must be in X. That is,

$$\forall x \in U[x \in X \Longleftarrow \exists y \in V(y \in X^{\Box} \wedge xRy)].$$

It can be re-expressed as a rule: for $x \in U$,

$$x \in X \Longleftarrow \bigvee_{y \in X^{\Box}} xRy. \tag{9}$$

The reverse implication does not hold in general. To derive a reverse implication, we construct another set of objects $X^{\Box\Diamond} \subseteq X$. For the set of objects, we have a rule: for $x \in U$,

$$x \in X^{\Box\Diamond} \Longrightarrow \bigvee_{y \in X^{\Box}} xRy. \tag{10}$$

This can be shown as follows:

$$\begin{aligned} x \in X^{\Box\Diamond} &\Longrightarrow xR \cap X^{\Box} \neq \emptyset \\ &\Longrightarrow \exists y \in V(xRy \wedge y \in X^{\Box}) \\ &\Longrightarrow \bigvee_{y \in X^{\Box}} xRy. \end{aligned} \tag{11}$$

In general, X is not the same as $X^{\Box\Diamond}$, which suggests that one can not establish a double implication rule for an arbitrary set of objects.

For a set of objects $X \subseteq U$, the pair $(X^{\Box\Diamond}, X^{\Box})$ is an object oriented formal concept. From the property $X^{\Box\Diamond\Box} = X^{\Box}$ and the rule (9), it follows:

$$x \in X^{\Box\Diamond} \Longleftarrow \bigvee_{y \in X^{\Box}} xRy. \tag{12}$$

By combining it with rule (10), we have a double implication rule:

$$x \in X^{\Box\Diamond} \Longleftrightarrow \bigvee_{y \in X^{\Box}} xRy. \tag{13}$$

The results can be extended to any object oriented formal concept. For $(X, Y) = (Y^{\Diamond}, X^{\Box})$, we have a rule:

$$x \in X \Longleftrightarrow \bigvee_{y \in Y} xRy. \tag{14}$$

That is, the set of objects X and the set of properties Y in (X, Y) uniquely determine each other.

B. The property oriented concept lattice

The property oriented concept lattice was introduced by Düntsch and Gediga [3]. A pair (X, Y), $X \subseteq U, Y \subseteq V$, is called a property oriented formal concept if $X = Y^{\Box}$ and $Y = X^{\Diamond}$. If a property is possessed by an object in X then the property must be in Y. Furthermore, only properties Y are possessed by objects in X.

The family of all property oriented formal concepts forms a lattice with meet $\wedge$ and join $\vee$ defined by:

$$\begin{aligned} (X_1, Y_1) \wedge (X_2, Y_2) &= (X_1 \cap X_2, (X_1 \cap X_2)^{\Diamond}) \\ &= (X_1 \cap X_2, (Y_1 \cap Y_2)^{\Box\Diamond}), \\ (X_1, Y_1) \vee (X_2, Y_2) &= ((Y_1 \cup Y_2)^{\Box}, Y_1 \cup Y_2) \\ &= ((X_1 \cup X_2)^{\Diamond\Box}, Y_1 \cup Y_2). \end{aligned}$$

For a set of objects $X \subseteq U$, we can construct a property oriented formal concept $(X^{\Diamond\Box}, X^{\Diamond})$. For a set of properties $Y \subseteq V$, there is a property oriented formal concept $(Y^{\Box}, Y^{\Box\Diamond})$.

For the formal context of Table I, the corresponding property oriented formal concept lattice is given by Figure 2.

Data analysis in the property oriented concept lattice can be carried out in the similar manner, but is focused on properties.

For a set of properties $Y \subseteq V$, we can construct a set of objects $Y^{\Box}$. It can be used to derive rules that determine whether a property is in Y. If a property is possessed by an object in $Y^{\Box}$, the property must be in Y. That is,

$$\forall y \in V[\exists x \in U(x \in Y^{\Box} \wedge xRy) \Longrightarrow y \in Y].$$

It can be re-expressed as a rule: for $y \in V$,

$$\bigvee_{x \in Y^{\Box}} xRy \Longrightarrow y \in Y. \tag{15}$$

In general, the reverse implication does not hold. In order to derive a reverse implication, we construct another set of

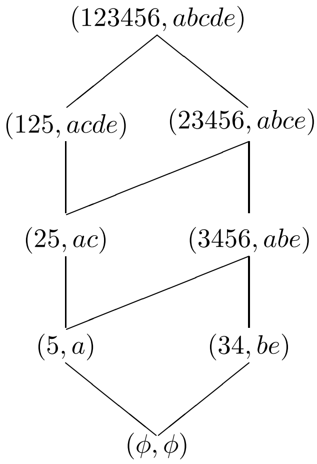

Fig. 2. A property oriented concept lattice

properties $Y^{\Box\Diamond} \subseteq Y$. For the set of properties, we have a rule: for $y \in V$,

$$\bigvee_{x \in Y^{\Box}} xRy \Longleftarrow y \in Y^{\Box\Diamond}. \tag{16}$$

Since X is not necessarily the same as $X^{\Box\Diamond}$, one can not establish a double implication rule for an arbitrary set of properties.

For a set of properties $Y \subseteq V$, the pair $(Y^{\Box\Diamond}, Y^{\Box})$ is a property oriented formal concept. From the property $Y^{\Box\Diamond\Box} = Y^{\Box}$ and the rule (15), it follows:

$$\bigvee_{x \in Y^{\Box}} xRy \Longrightarrow y \in Y^{\Box\Diamond}. \tag{17}$$

By combining it with rule (16), we have a double implication rule:

$$\bigvee_{x \in Y^{\Box}} xRy \Longleftrightarrow y \in Y^{\Box\Diamond}. \tag{18}$$

The results can be extended to any property oriented formal concept. For $(X, Y) = (Y^{\Box}, X^{\Diamond})$, we have a rule:

$$\bigvee_{x \in X} xRy \Longleftrightarrow y \in Y. \tag{19}$$

That is, the set of properties Y and the set of objects X in (X, Y) uniquely determine each other.

C. Connections of the two lattices

By the duality of approximation operators $^{\Box}$ and $^{\Diamond}$, it can be easily seen that the object oriented concept lattice is isomorphic to the property oriented concept lattice. Given an object oriented concept:

$$(X, Y)_o = (Y^{\Diamond}, X^{\Box})_o, \tag{20}$$

we can obtain a property oriented concept:

$$\begin{aligned} (X^c, Y^c)_p &= (Y^{\Diamond c}, X^{\Box c})_p \\ &= ((Y^c)^{\Box}, (X^c)^{\Diamond})_p. \end{aligned} \tag{21}$$

Conversely, from a property oriented concept one can obtain an object oriented concept. For example, the object oriented concept $(\{1\}, \{d\})_o$ corresponds to the property oriented concept $(\{2, 3, 4, 5, 6\}, \{a, b, c, e\})_p$, and vice versa. However, as shown earlier, their semantic interpretations are different. One is used to carry out object oriented inference, and the other is used to carry out property oriented inference. From the semantic viewpoint, it is meaningful to study both lattices [3].

For an easy comparison of the two lattices, we summary the main results as follows. An object oriented concept $(X, Y)_o = (Y^{\Diamond}, X^{\Box})_o$ is characterized by a rule of the form:

$$x \in X \Longleftrightarrow \bigvee_{y \in Y} xRy. \tag{22}$$

On the other hand, a property oriented concept $(X, Y)_p = (Y^{\Box}, X^{\Diamond})_p$ is characterized by another rule of the form:

$$\bigvee_{x \in X} xRy \Longleftrightarrow y \in Y. \tag{23}$$

Several observations can be made. The two types of concepts of the two lattices share a common feature, that is, each concept consists of a subset of objects and a subset of properties that determine each other. Their differences lie in the representations of the involved subsets of objects and properties. In the object oriented concept lattice, one uses a subset of properties to construct a condition to describe a subset of objects. In the property oriented lattice, the reverse is true, namely, one uses a subset of objects to construct a condition to describe a subset of properties. Furthermore, the conditions are all expressed in terms of disjunction.

D. Relationships to formal concept lattice

One can establish a relation between the two lattices and the standard formal concept lattice [3], [10].

For an object oriented concept $(X, Y) = (Y^{\Diamond}, X^{\Box})$, we can re-express rule (22) as follows:

$$x \in X^c \Longleftrightarrow \bigwedge_{y \in Y} \neg(xRy). \tag{24}$$

It is another kind of double implication with respect to the pair (X^c, Y). The rule suggests that if an object does not have all properties in Y, then the object does not belong to X, and vice versa. Clearly, we have $(X^c, Y) = (Y^{\Diamond c}, X^{cc\Box})$. By setting X^c as Z, we have a pair $(Z, Y) = (Y^{\Diamond c}, Z^{c\Box})$.

Based on the above discussion, we can define a new type of formal concepts. A pair (X, Y), $X \subseteq U$, $Y \subseteq V$, is called a formal concept if $X = Y^{\Diamond c}$ and $Y = X^{c\Box}$. By the duality of $^{\Box}$ and $^{\Diamond}$, we have:

$$\begin{aligned} X &= Y^{\Diamond c} = Y^{c\Box} = X^{\Diamond\Box}, \\ Y &= X^{c\Box} = X^{\Diamond c} = Y^{\Diamond\Box}. \end{aligned} \tag{25}$$

That is, (X, Y) is a formal concept if and only $X = Y^{c\Box}$ and $Y = X^{c\Box}$. Let us denote the operator $^{c\Box}$ by *. Then, (X, Y) is a formal concept if and only $X = Y^*$ and $Y = X^*$.

TABLE II

THE COMPLEMENT FORMAL CONTEXT OF TABLE I

	a	b	c	d	e
1		×			
2		×		×	×
3	×		×	×	
4	×		×	×	
5		×	×	×	×
6			×	×	

Suppose $(X, Y) = (Y^{c\Box}, X^{c\Box})$ is a formal concept. Consider the pair $(Y^{\Diamond}, Y)$, we have:

$$\begin{aligned}(Y^{\Diamond}, Y) &= (Y^{\Diamond}, X^{c\Box}) \\ &= (Y^{\Diamond}, Y^{\Diamond\Box}). \end{aligned} \tag{26}$$

Thus, $(Y^{\Diamond}, Y)$ is an object oriented concept. Similarly, the pair,

$$\begin{aligned}(X, X^{\Diamond}) &= (Y^{c\Box}, X^{\Diamond}) \\ &= (X^{\Diamond\Box}, X^{\Diamond}), \end{aligned} \tag{27}$$

is a property oriented concept. That is, a formal concept takes the set of objects from an object oriented concept, and the set of properties from a related property oriented concept.

For a formal concept,

$$\begin{aligned}(X, Y) &= (Y^{\Diamond c}, X^{c\Box}) \\ &= (Y^{c\Box}, X^{\Diamond c}), \end{aligned} \tag{28}$$

we have a pair of rule:

$$\begin{aligned} x \in X &\iff x \in Y^{\Diamond c} \\ &\iff x \notin Y^{\Diamond} \\ &\iff xR \cap Y = \emptyset \\ &\iff \bigwedge_{y \in Y} \neg(xRy); \end{aligned} \tag{29}$$

$$\begin{aligned} \bigwedge_{x \in X} \neg(xRy) &\iff Ry \cap X = \emptyset \\ &\iff Ry \subseteq X^c \\ &\iff y \in X^{c\Box} \\ &\iff y \in Y. \end{aligned} \tag{30}$$

In rule (22), a set of objects is characterized based on the *presence* of *at least one* property in a set of properties. In the new rule (29), a set objects is characterized based on the *absence* of *all* properties in a set of properties. Similar observation can be made regarding rules (23) and (30).

The family of all formal concepts forms a lattice with the meet $\wedge$ and the join $\vee$ defined as follows:

$$\begin{aligned} (X_1, Y_1) \wedge (X_2, Y_2) &= (X_1 \cap X_2, (X_1 \cap X_2)^{c\Box}) \\ &= (X_1 \cap X_2, (Y_1 \cup Y_2)^{\Diamond\Box}), \\ (X_1, Y_1) \vee (X_2, Y_2) &= ((Y_1 \cap Y_2)^{c\Box}, Y_1 \cap Y_2) \\ &= ((X_1 \cup X_2)^{\Diamond\Box}, Y_1 \cap Y_2). \end{aligned}$$

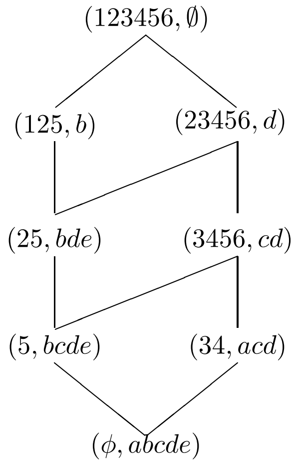

Fig. 3. The formal concept lattice of the formal context in Table II

This lattice is isomorphic to both object oriented and property oriented concept lattices. For the new lattice, we define the sub-concept relationship as follows. A concept (X_1, Y_1) is sub-concept of (X_2, Y_2) if only if $X_1 \subseteq X_2$, or equivalently, if and only if $Y_1 \supseteq Y_2$. That is, a smaller set of objects is characterized by a larger set of properties.

For a binary relation R, its complement relation is defined by:

$$R^c = \{(x, y\} \mid \neg(xRy)\}. \tag{31}$$

The formal context (U, V, R^c) is referred to as the complement formal context of (U, V, R). The approximation operator $^\Box$ in (U, V, R) and the derivation operator of formal concept lattice of (U, V, R^c) determines each other [3], [10]. The lattice of concepts (X, Y) defined by condition $X = Y^{c\Box}$ and $Y = X^{c\Box}$ is the formal concept lattice of the formal context (U, V, R^c).

An important difference between object oriented concepts, property oriented concepts and formal concepts of formal concept analysis is that the first two use disjunction in forming a condition, while the last one uses conjunction [10]. Each of them captures a particular aspect of knowledge embedded in a formal context.

The complement formal context of Table I is given in Table II. The corresponding concept lattice is given in Figure 3. It is a combination of the object oriented concept lattice in Figure 1 and the property oriented concept lattice in Figure 2. All three lattices are isomorphic to each other.

E. Summary

The main results of the three types of concepts and the corresponding concept lattices are summarized below.

- Object oriented concept lattice
  - Definition:
    (X, Y) is an object oriented concept if and only if $X = Y^{\Diamond}$ and $Y = X^{\Box}$.

– Lattice operations:

$$(X_1, Y_1) \wedge (X_2, Y_2) = ((X_1 \cap X_2)^{\square\Diamond}, Y_1 \cap Y_2),$$
$$(X_1, Y_1) \vee (X_2, Y_2) = (X_1 \cup X_2, (Y_1 \cup Y_2)^{\Diamond\square}).$$

– Inference rule: for all $x \in U$,

$$x \in X \Longleftrightarrow \bigvee_{y \in Y} xRy.$$

- Property oriented concept lattice
  – Definition:
  (X, Y) is a property oriented concept if and only if $X = Y^{\square}$ and $Y = X^{\Diamond}$.
  – Lattice operations:

$$(X_1, Y_1) \wedge (X_2, Y_2) = (X_1 \cap X_2, (Y_1 \cap Y_2)^{\square\Diamond}),$$
$$(X_1, Y_1) \vee (X_2, Y_2) = ((X_1 \cup X_2)^{\Diamond\square}, Y_1 \cup Y_2).$$

  – Inference rule: for all $y \in V$,

$$\bigvee_{x \in X} xRy \Longleftrightarrow y \in Y.$$

- Formal concept lattice of the formal context (U, V, R^c)
  – Definition:
  (X, Y) is a formal concept if and only if $X = Y^{c\square}$ and $Y = X^{c\square}$.
  – Lattice operations:

$$(X_1, Y_1) \wedge (X_2, Y_2) = (X_1 \cap X_2, (Y_1 \cup Y_2)^{\Diamond\square}),$$
$$(X_1, Y_1) \vee (X_2, Y_2) = ((X_1 \cup X_2)^{\Diamond\square}, Y_1 \cap Y_2).$$

  – Inference rules: for all $x \in U$ and $y \in V$,

$$x \in X \iff \bigwedge_{y \in Y} \neg(xRy),$$
$$\bigwedge_{x \in X} \neg(xRy) \iff y \in Y.$$

The relationships between the three types of concepts are summarized as follows.

- If $(X, Y) = (Y^{\Diamond}, X^{\square})$ is an object oriented concept, then
  – (X^c, Y^c) is a property oriented concept,
  – $(X^c, Y) = (Y^{c\square}, Y)$ is a formal concept of the formal context (U, V, R^c).
- If $(X, Y) = (Y^{\square}, X^{\Diamond})$ is a property oriented concept, then
  – (X^c, Y^c) is an object oriented concept,
  – $(X, Y^c) = (X, X^{c\square})$ is a formal concept of the formal context (U, V, R^c).
- If $(X, Y) = (Y^{c\square}, X^{c\square})$ is a formal concept of the formal context (U, V, R^c), then
  – $(X^{c\square\Diamond}, Y) = (Y^{\Diamond}, Y)$ is an object oriented concept,
  – $(X, Y^{c\square\Diamond}) = (X, X^{\Diamond})$ is a property oriented concept.

Each type of concepts captures a particular aspect of data. The three concept lattices are related, but different from each other. They collectively provide additional tools for data analysis.

IV. Conclusion

There are at least three possible directions in comparing and combining the theory of rough sets and formal concept analysis. One is the introduction of the notions of rough set theory, for example, approximation operators, into formal concept analysis. The second option is the introduction of the notions of concept analysis into rough set theory. The third option to combine the two theories into a common, and more general, framework.

This paper deals with the introduction of the notions of formal concept analysis to rough set theory, which leads to new, different interpretations and representations of formal concepts. Based on the approximation operators, three concept lattices are introduced and examined. They are different from and related to formal concepts in formal concept analysis. Rough set theory focuses on the disjunctive description of concepts, while formal concept analysis focuses on conjunctive description of concepts. The combination of the two views may bring more insights into data analysis.

References

[1] Düntsch, I. and Gediga, G. Approximation operators in qualitative data analysis, in: *Theory and Application of Relational Structures as Knowledge Instruments*, de Swart, H., Orlowska, E., Schmidt, G. and Roubens, M. (Eds.), Springer, Heidelberg, 216-233, 2003.

[2] Ganter, B. and Wille, R. *Formal Concept Analysis, Mathematical Foundations*, Springer, Berlin, 1999.

[3] Gediga, G. and Düntsch, I. Modal-style operators in qualitative data analysis, *Proceedings of the 2002 IEEE International Conference on Data Mining*, 155-162, 2002.

[4] Hu, K., Sui, Y., Lu, Y., Wang, J. and Shi, C. Concept approximation in concept lattice, *Knowledge Discovery and Data Mining, Proceedings of the 5th Pacific-Asia Conference, PAKDD 2001, Lecture Notes in Computer Science 2035*, 167-173, 2001.

[5] Kent, R.E. Rough concept analysis: a synthesis of rough sets and formal concept analysis, *Fundamenta Informaticae*, **27**, 169-181, 1996.

[6] Saquer, J. and Deogun, J.S. Formal rough concept analysis, *New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, 7th International Workshop, RSFDGrC '99, Lecture Notes in Computer Science 1711*, Springer, Berlin, 91-99, 1999.

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

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

[9] Wille, R. Restructuring lattice theory: an approach based on hierarchies of concepts, in: *Ordered Sets*, Rival, I. (Ed.), Reidel, Dordrecht-Boston, 445-470, 1982.

[10] Yao, Y.Y. A comparative study of formal concept analysis and rough set theory in data analysis, manuscript, 2004.

[11] Yao, Y.Y. and Y. Chen, Rough set approximation in formal concept analysis, manuscript, 2004.

[12] Yao, Y.Y. and Lin, T.Y., Generalization of rough sets using modal logic, *Intelligent Automation and Soft Computing, An International Journal*, **2**, 103-120, 1996.

[13] Yao, Y.Y., Wong, S.K.M. and Lin, T.Y. A review of rough set models, in: *Rough Sets and Data Mining: Analysis for Imprecise Data*, Lin, T.Y. and Cercone, N. (Eds.), Kluwer Academic Publishers, Boston, 47-75, 1997.