Knowledge dependency degree in TRSM and its application to robot rod catching control

Chen Wu, Wei Zhu, Lijuan Wang

***Abstract*—**Through extending the related concept from complete decision table to incomplete one, the present paper first defines the concept of complete knowledge dependency and discusses relationships between tolerance class and indispensable attribute and knowledge dependency. It proves that reflectivity, transitivity, augmentation, decomposition rule and merge rule are valid for complete knowledge dependency. Secondly, it newly defines dependency degree and partial dependency degree in incomplete decision table with respect to whether or not decision attribute exist missing decision value. By finding that partial dependency degrees after transferring, augmenting and decomposing are not always kept in the same, it reveals several laws with proved theorems. Finally, it uses the knowledge dependency and dependency degree to design an algorithm to solve attribute reduction of incomplete decision table and apply the algorithm to realize robot rod catching control problem solving.

***Keywords*—**attribute reduct, knowledge dependency, incomplete information system, robot control, rough set.

I. Introduction

USING data set itself, rough set discovers potentially hidden knowledge from large data, especially, from complete information systems or decision tables, by introducing indiscernbility relation and lower and upper approximations [1][2][3]. So, rough set, as a useful mathematics tool, has already applied in many scientific areas such as pattern recognition, data mining, machine learning, decision making, and etc. Unlike in complete information systems, an indiscernbility relation or equivalence relation, based on attribute value comparisons, may be easily built to find equivalence classes as knowledge granules to look into indiscernibility in complete information systems, such an indiscernbility relation or equivalence relation may not be directly obtained because of that some attribute values of objects are probably missing or not observed in incomplete information system (IIS) or incomplete decision table (IDT) due to some limitations of real world data measurement or data acquirement. Completeness is relative, but incompleteness is absolute. If we can invent some approaches to cope with incomplete information systems, we will be powerful to deal with large amount of information systems because complete information systems are special cases of incomplete ones. Realized such a importance reason, scientists have already been transferred to investigate incomplete information systems from complete information systems in recent years.

There has already exists two main kinds of approaches in handling incomplete information systems now. One first substitutes a missing data with the most frequent appeared value of the relevant attribute values or with the mean value of them to make an incomplete system complete, from the view of statistics, and then to analyze it as a complete system. This is called indirect approach. The other lets any missing data in incomplete information system non-replaced. In this approach, concepts are appropriately generated from complete system to incomplete one. It is also called direct approach. However, indiscernbility relations or equivalence relations are no longer used. Tolerance relation, similarity relation and other binary relations, not being equivalence relations, are introduced. Equivalence classes are also replaced by using tolerance class, similarity class or other classes simultaneously. Direct processing approach also brings about many advantages for it avoids some subjective replacements. Many scientists are interest in direct approach and many new rough set models and results are studied out.

Based on two different comprehensions about missing value in IIS or IDT, corresponding concrete methods or models are put forward. One comprehension about missing value is that it is lost currently for some reason but it really exists. The other is that missing value is not present for some reason such as privacy. It should be kept absent and prohibited to make comparisons with other values.

Among generalized rough set models coping with incomplete information systems directly, following models or methods are representative and have big influences in rough set areas: tolerance relation rough set model, put forward in [4], similarity relation, introduced in [5][6], limited tolerant relation, proposed in [7], maximal consistent block technique, suggested in [8][9], different information granule models, proposed in [10], compatibility relation rough set model, discussed in [11], variable precision rough set models, discussed in many literatures [12].

In tolerance rough set model, instead of equivalence class, tolerance class is applied. Generalized decision function is

This This work was supported in part by the China NFS (No. 61100116) and a Foundation of Graduate Department of Jiangsu Province and Jiangsu University of Science and Technology(JGLX17_085).

Chen Wu is with the Jiangsu University of Science and Technology, Zhenjiang City, Jiangsu Province, China (phone: (+86)13861390697; e-mail: wuchenzj@sina.com).

Wei Zhu and Lijuan Wang are all with the Jiangsu University of Science and Technology (e-mail: 179070001@qq.com, yzlujing@sina.com, respectively).

suggested to solve knowledge reduction problem. Similarity relation is used to process non-symmetric relation problem. Limited tolerance relation is with a stronger strict condition than tolerance relation. Maximal consistent block technique is introduced for rule acquisition. Different information granule models are based on granules from general and complete covering. Compatibility relation rough set model focuses on compatibility. Variable precision rough set models expand conclusion to approximate conclusion to handle imprecision.

Some algorithms to find approximation sets of a given object set are designed in [13] through using binary matrices. Multi-granular rough set theory is discussed in [14]. Generalized rough set approaches are used for processing incomplete information system by scientists in [9][15]. Applications of generalized rough set models make them useful [16]. Decision rough set models as new research areas have already commenced to become new attractive topics [17].

Based on the first semantic comprehension about missing value in IIS or IDT, the present paper first studies some properties of incomplete information system in order to clear up some concepts different from complete information system. Then it introduces the extended definition of complete knowledge dependency in [18] from complete information system and obtains that complete knowledge dependency holds some preserving laws such as reflexivity, transitivity, augmentation, decomposition and merge on attribute subsets or tolerance relations in incomplete information system. Through exploring partial knowledge dependency and dependency degree, it proves that partial knowledge dependency degree possesses some special relationships on transitivity, augmentation, and decomposition of attribute subsets or tolerance relations.

It also designs an algorithm to find reductions of an incomplete information system or incomplete decision table based on knowledge dependency. Using knowledge dependency degree and the algorithm to find attribute reduction of an incomplete system, it solves robot rod catching control problem as an example. The work, as the title called “knowledge dependency degree in tolerance rough set model and its application to robot rod catching control”, helps us further clear up some properties in incomplete information system or incomplete decision table, so it has some importance.

II. Basic concepts

An incomplete information system or incomplete decision table is a quadruple $IIS=(S,AT,V,f)$[4], where S is the finite non-empty universe of objects; $AT=C\cup D$ is a finite non-empty attribute set. C is the condition attribute set. D is the decision attribute set. $C\cap D=\varnothing$. If $D=\varnothing$, the IIS is called incomplete information system. If $D\neq\varnothing$, the IIS is called incomplete decision table. Without loss of generality we call incomplete information system referring both of them. For any $a\in AT$, V_a is called the value universe of attribute a, $a:S\rightarrow V_a$ is a mapping such that $f(x,a)\in V_a$ for any object x in S. For any object, some one of its attribute values may be lost or missing, called a null value (a null is denoted by *). So for any $a\in AT, *\in V_a$ may be true. $V=\cup V_a(a\in AT)$ is called the set of all attribute values.

Definition 1. Let $P\subseteq AT$ be any attribute subset in $IIS=(S,AT,V,f)$. A tolerance relation on S, denoted by $R(P)$, is defined as follows:

$$R(P)=\{(x,y)\mid \forall a\in P(f(x,a)=f(y,a)\vee f(x,a)=* \vee f(y,a)=*)\}. \quad (1)$$

If $P=\{a\}$ is a singleton attribute subset, then $R(P)$ is abbreviated to be $R(a)$.

$R(P)$ is of reflexivity and symmetry on S.

Definition 2. For $\forall P\subseteq AT$,

$$S/R(P)=\{X \mid X^2\subseteq R(P),\ \forall y\notin X,\ (X\cup\{y\})^2\not\subseteq R(P)\} \quad (2)$$

is called a compatible knowledge system on S since each element in $S/R(P)$ as a granule is mutually compatible with the relation $R(P)$. It is also called a complete cover, denoted by S/P for short.

Definition 3. For $\forall P\subseteq AT, x\in S$,

$$U_P(x)=\cup B(B\in S/R(P), x\in B\} \quad (3)$$

It has already been proved that $R(P)=R(P)$ and $U_P(x)=S_P(x)$ in [4[[13]. $U_P(x)$ has an intuitive meaning and a similar computation method to $S_P(x)$. $U_P(x)$ is called a tolerance class generated by x as a generator, abbreviated to be $U_a(x)$ if $P=\{a\}$ is a singleton set.

Definition 4. Let $X\subseteq S$, $P\subseteq AT$, the lower and the upper approximations of X are respectively defined by

$$P\_(X)=\{y|y\in S, U_P(y)\subseteq X\}, \quad (4)$$

$$P^-(X)=\{y|y\in S, U_P(y)\cap X\neq\varnothing\}. \quad (5)$$

Definition 5. Let $P\subseteq AT$, $P\neq\varnothing$, $a\in P$. a is dispensable in P if, and only if

$$R(P)=R(P\text{-}\{a\}), \quad (6)$$

P is independent if, and only if each $a\in P$ is indispensable. Otherwise, P is dependent.

Theorem 1. Suppose $a\in P$ is dispensable in P. If $b\in P$ and for any $x\in S$, $U_a(x)=U_b(x)$, then b is also dispensable in P.

Proof. Since $a\in P$ is dispensable, we have $R(P)=R(P\text{-}\{a\})$. For any $x\in S$, $U_{P\text{-}\{a\}}(x)=U_P(x)$. $U_P(x)=\cap U_c(x)(c\in P)=\cap U_c(x)$ $(c\in P\text{-}\{a\})=\cap U_c(x)(c\in P\text{-}\{b\})=U_{P\text{-}\{b\}}(x)$. It follows that $R(P)=R(P\text{-}\{b\})$, so b is also dispensable.

Definition 6. Q is called a reduction of P if, and only if $Q\subseteq P, R(Q)=R(P)$ and Q is independent.

Definition 7. $core(P)$ is called the core of P if $core(P)$ is a set that contains all indispensable attribute of P:

$$core(P)=\cap Q(Q\in red(P)), \quad (7)$$

where $red(P)$ is the collection of all reductions of P. This reflects the relationship between the core and reductions of P.

Theorem 2. Let $P\subseteq A$. If for $a\in P$ and any $x\in S$,

$$U_a(x)=\cup U_{P\text{-}\{a\}}(y)\ (y\in U_a(x)), \quad (8)$$

then a is dispensable in P, i.e. $R(P\text{-}\{a\})=R(P)$.

Proof. Since $a\in P$, $R(P)\subseteq R(P\text{-}\{a\})$. Hence, we need only prove $R(P\text{-}\{a\})\subseteq R(P)$. For any $(x,y)\in R(P\text{-}\{a\})$, we have $y\in U_{P\text{-}\{a\}}(y)\subseteq \cup U_{P\text{-}\{a\}}(y)\ (y\in U_a(x))=U_a(x)$. Thus, $(x,y)\in R(a)$. It follows that $R(P\text{-}\{a\})\subseteq R(a)$. Therefore, $R(P)=$

$R(P\text{-}\{a\}) \cap R(a) \supseteq R(P\text{-}\{a\})$. In other word, $R(P\text{-}\{a\}) \subseteq R(P)$. Thus, $R(P\text{-}\{a\})= R(P)$.

III. Knowledge Dependency in IIS

In complete information system, by what b is dependent on a (a and b are two attributes), denoted by $a \Rightarrow b$, is defined as if $f(x_1,a)=f(x_2,a)$ for $\forall\ x_1,\ x_2 \in S,\ x_1 \neq x_2$, then it must have $f(x_1,b)=f(x_2,b)$. But in incomplete information system, such a definition is no longer valid because of existing missing values. A newly defined knowledge dependency (KD for short) in incomplete information system is given as follows.

Definition 8. In an IIS, let $a, b \in AT$. $a \Rightarrow b$ if, and only if for $\forall\ x_1, x_2 \in S$,

$$x_1 \neq x_2, f(x_1,a)=f(x_2,a) \vee f(x_1,a)=* \vee f(x_2,a)=*, \tag{9}$$

then

$$f(x_1,b)= f(x_2,b) \vee f(x_1,b)=* \vee f(x_2,b)=*. \tag{10}$$

This definition can be viewed as an extension of knowledge dependency in complete information system to incomplete information.

We can prove the following two results:

Theorem 3. $a \Rightarrow b$ if, and only if for $\forall\ x \in S$,

$$U_a(x) \subseteq U_b(x). \tag{11}$$

Theorem 4. Let $a, b \in AT$. $a \Rightarrow b$ if, and only if

$$R(a) \subseteq R(b). \tag{12}$$

Definition 9. Let $P,Q \subseteq AT$. $P \Rightarrow Q$ if, and only if $p \Rightarrow q$ for any $p \in P$ and $q \in Q$.

The following theorem can also be obtained.

Theorem 5. $P \Rightarrow Q$, if, and only if

$$R(P) \subseteq R(Q). \tag{13}$$

If $P \Rightarrow Q$, then we call Q is dependent on P or P determines Q.

Theorem 6. Let $a \in P$. a is dispensable in P if

$$R(P\text{-}\{a\}) \subseteq R(P). \tag{14}$$

Proof. Since $P\text{-}\{a\} \subseteq P$, we have $R(P) \subseteq R(P\text{-}\{a\})$. Now $R(P\text{-}\{a\}) \subseteq R(P)$, so $R(P\text{-}\{a\})=R(P)$, i.e. a is dispensable in P.

Theorem 7. Let $a \in P$. If

$$P\text{-}\{a\} \Rightarrow P, \tag{15}$$

then a is dispensable in P.

Proof. Because $P\text{-}\{a\} \Rightarrow P$, thus $R(P\text{-}\{a\}) \subseteq R(P)$. Since $P\text{-}\{a\} \subseteq P$, we have $R(P) \subseteq R(P\text{-}\{a\})$. Therefore $R(P\text{-}\{a\})=R(P)$, i.e. a is dispensable in P.

Theorem 8. Let $a \in P$. If

$$R(P\text{-}\{a\})=R(a), \tag{16}$$

then a is dispensable in P.

Proof. Because $R(P\text{-}\{a\})=R(a)$, we have $R(P)=R(P\text{-}\{a\}) \cap R(a)=R(a)$. Thus $R(P\text{-}\{a\})=R(P)$, i.e. a is dispensable in P.

Theorem 9. Let $a \in P$. If $P\text{-}\{a\} \Rightarrow a$ and $a \Rightarrow P\text{-}\{a\}$, then a is dispensable in P.

Proof. Because $P\text{-}\{a\} \Rightarrow a$ and $a \Rightarrow P\text{-}\{a\}$, we have, $R(P\text{-}\{a\}) \subseteq R(a)$ and $R(a) \subseteq R(P\text{-}\{a\})$, that is $R(P\text{-}\{a\})=R(a)$. According to the above theorem, a is dispensable in P.

According to the relative definitions in complete information system, the following expressions are equivalent:

(i) $P \Rightarrow Q$; (17)

(ii) $R(P \cup Q)= R(P)$; (18)

(iii) $POS_P(Q)=S$. (19)

But in incomplete information system, the situation may not be always valid.

Theorem 10. Only the following two expressions are equivalent to incomplete information system:

(i) $P \Rightarrow Q$; (20)

(ii) $R(P \cup Q)= R(P)$. (21)

Proof. First prove (i)⇒(ii). Because $P \Rightarrow Q$, we have $R(P) \subseteq R(Q)$. Therefore, $R(P \cup Q) = R(P) \cap R(Q)= R(P)$. So, (ii) holds.

Then we prove (ii)⇒(i). Because $R(P \cup Q)=R(P)$, and $R(P \cup Q)=R(P) \cap R(Q)$, we have, $R(P) \cap R(Q)= R(P)$. Thus $R(P) \subseteq R(Q)$, that is $P \Rightarrow Q$.

Therefore, (i) and (ii) are equivalent.

Theorem 11. If $P \Rightarrow Q$, $P \subseteq Q$, then

$$POS_P(Q)= \cup P\_(X)(X \in S/R(Q))=S. \tag{22}$$

Proof. It is obvious that $POS_P(Q) \subseteq S$. So we need only prove $S \subseteq POS_P(Q)$. Because $P \Rightarrow Q$ and $P \subseteq Q$, therefore $R(P) \subseteq R(Q)$, $R(Q) \subseteq R(P)$. Thus, $R(P)=R(Q)$. Therefore, $U_P(y)= U_Q(y)$ for any $y \in S$.

For $\forall\ y \in S, U_Q(y) \in S/R(Q)$, $y \in U_Q(y) \subseteq \cup \{y \in S|U_P(y) =U_Q(y) \subseteq U_Q(y)\}(U_Q(y) \in S/R(Q)) =POS_P(Q)$, we have $y \in POS_P(Q)$. Because $y \in U$ is arbitrarily chosen, $S \subseteq POS_P(Q)$. So $POS_P(Q)= \cup P\_(X)(X \in S/R(Q))=S$.

Notice that $POS_P(Q)= \cup P\_(X)(X \in S/R(Q))=S$ does not mean $P \Rightarrow Q$.

IV. Relations between KD and Tolerance Classes

Theorem 12. Let $a \in AT$, $P \subseteq AT$, $\forall\ x \in S$. If $P \Rightarrow a$, then

$$U_a(x) \subseteq \cup U_P(y)(y \in U_a(x)). \tag{23}$$

Proof. For $\forall\ z \in U_a(x)$, $z \in U_P(z) \subseteq \cup U_P(y)(y \in U_a(x))$, we have $U_a(x) \subseteq \cup U_P(y)(y \in U_a(x))$.

Theorem 13. Let $P \subseteq AT$. If

$$U_a(x)= \cup U_P(y)(y \in U_a(x)), \tag{24}$$

for $\forall\ x \in S$, then $P \Rightarrow a$.

Proof. Because a necessary and sufficient condition of $P \Rightarrow a$ is $R(P) \subseteq R(a)$, we need only to prove $R(P) \subseteq R(a)$.

For any $(x,y) \in R(P)$, we have $y \in U_P(x)$. Because $x \in U_a(x)$, $y \in U_P(x) \subseteq \cup U_P(y)\ (y \in U_a(x))=U_a(x)$, i.e.,$y \in U_a(x)$. So, $(x, y) \in R(a)$. For $(x, y) \in R(P)$ is arbitrarily chosen, we have $R(P) \subseteq R(a)$.

The above conclusion holds whether $a \in P$ or not.

Theorem 14. If $a \in P$ and $U_a(x)= \cup U_P(y)(y \in U_a(x))$ for $\forall\ x \in S$, then a is dispensable in P.

Proof. According to the above theorem, we know $P\text{-}\{a\} \Rightarrow a$, i.e. $R(P\text{-}\{a\}) \subseteq R(a)$. Because again $R(P) \subseteq R(P\text{-}\{a\})$, therefore we only need to prove $R(P\text{-}\{a\}) \subseteq R(P)$.

For any $(x,y) \in R(P\text{-}\{a\})$, we have $y \in U_{P\text{-}\{a\}}(x)$. Because $x \in U_a(x)$, thus, $y \in U_{P\text{-}\{a\}}(x) \subseteq \cup U_P(y)(y \in U_a(x))=U_a(x)$, i.e. $y \in U_a(x)$. So, $(x,y) \in R(a)$. Thus, $(x,y) \in R(P\text{-}\{a\}) \wedge (x,y) \in R(a)$. Therefore, $(x,y) \in R(P\text{-}\{a\}) \cap R(a) =R(P)$. It follows that $R(P\text{-}\{a\}) \subseteq R(P)$. Summing up, $R(P\text{-}\{a\})=R(P)$. So, a is dispensable in P.

$P \Rightarrow Q$ does not mean $P\_(X)=X$ for any $X \in S/R(Q)$.

$P \Rightarrow a$ and $a \in P$ do not imply that a is dispensable in P, i.e. $P \Rightarrow a$ and $a \in P$ do not mean $R(P\text{-}\{a\})=R(P)$.

Table I. An IIS of robot catching rod

S	a	b	c	d	e	f
1	0	0	*	*	0	1
2	0	*	1	0	*	1
3	1	0	*	*	0	1
4	*	*	0	1	0	2
5	0	*	1	*	0	2
6	1	*	1	0	*	3
7	1	1	*	*	*	4

Example 1. An incomplete information system about robot rod catching problem is shown in table I, where a, b, c, d, e are condition attributes, f is decision attribute. Means of them are to be explained in detail in section 6.

Let $AT=\{a, b, c, d, e\}$,$A=\{a, b, c, d\}$. $R(AT)= R(A)$. $U_{AT}(1)=U_A(1)=\{1,2,4,5\}$, $U_{AT}(2)=U_A(2) =\{1,2, 5\}$, $U_{AT}(3)=U_A(3)= \{3,4,6\}$, $U_{AT}(4)=U_A(4) =\{1,2,4,7\}$, $U_{AT}(5)=U_A(5)=\{1,2,5\}$, $U_{AT}(6)=U_A(6) =\{3,6\}$, $U_{AT}(7)= U_A(7) =\{4,6,7\}$. AT is functional dependent on A. It can be checked that A is a reduction of AT. But for any $X \in S/AT$, $A\_(X)=X$ may not always hold. For instance, $A\_(\{1,2,4,5\}) =\{1,2,5\} \neq \{1,2,4,5\}$, $A\_(\{3,4, 6\}) =\{3,6\} \neq \{3,4,6\}$, $A\_(\{1,2,4,7\}) =\{4\} \neq \{1,2,4,7\}$, $A\_(\{1, 2,5\})=\{2,5\} \neq \{1,2,5\}$, $A\_(\{3,6\}) =\{6\} \neq \{3,6\}$. $A\_(\{4,6,7\})= \varnothing \neq \{4,6,7\}$. Because A is a reduction of AT, e is dispensable in AT, $AT \Rightarrow e$.

In Table I, $U_e(1)=\{1,2,3,4,5,6,7\}$, $U_{AT\text{-}\{e\}}(1)=\{1,2,4,5\}$,$U_{AT\text{-}\{e\}}(2)=\{1,2,5\}$, $U_{AT\text{-}\{e\}}(3)=\{3,4,6\}$, $U_{AT\text{-}\{e\}}(4)=\{1,3,4,7\}$, $U_{AT\text{-}\{e\}}(5)=\{1,2,5\}$, $U_{AT\text{-}\{e\}}(6)=\{3,6\}$, $U_{AT\text{-}\{e\}}(7)=\{4,6,7\}$, $U_e(6)=\{2,6,7\} \neq \cup U_{AT\text{-}\{e\}}(y)(y \in U_e(6))=\{1,2,3,4,5,6,7\}$. $U_e(7)= \{2,6,7\} \neq \cup U_{AT\text{-}\{e\}}(y) (y \in U_e(7))=\{1,2,3,4,5,6,7\}$. We have only $U_e(6)= \{2,6,7\} \subset \cup U_{AT\text{-}\{e\}}(y)(y \in U_e(6))=\{1,2,3,4, 5,6,7\}$. $U_e(7)=\{2,6,7\} \subset \cup U_{AT\text{-}\{e\}}(y)(y \in U_e(7)) =\{1,2,3,4,5, 6,7\}$.

Theorem 15. Let $P, Q, R, T \subseteq AT$. Then the following laws are valid.

(i) if $Q \subseteq P \subseteq AT$, then $P \Rightarrow Q$. (Reflexivity law)

(ii) if $P \Rightarrow Q$ and $Q \Rightarrow R$, then $P \Rightarrow R$. (Transitivity law)

(iii) if $P \Rightarrow Q$ and $Q \Rightarrow R$, then $P \cup Q \Rightarrow R$. (Left merge law)

(iv) if $P \Rightarrow Q \cup R$, then $P \Rightarrow Q$ and $P \Rightarrow R$. (Decomposition law)

(v) if $P \to Q$ and $Q \cup R \Rightarrow T$, then $P \cup R \Rightarrow T$. (Pseudo Transitivity law)

(vi) if $P \Rightarrow Q$ and $R \Rightarrow T$, then $P \cup R \Rightarrow Q \cup T$. (Merge law)

(vii) if $P \Rightarrow Q$ and $P \subseteq R$, then $R \Rightarrow Q$. (Augmentation law)

In the above, (iv) can be re-expressed in an equivalent form:

(iv') if $P \Rightarrow Q$ and $R \subseteq Q$, then $P \Rightarrow R$. (Decomposition law)

Simultaneously, (vii) can be re-expressed in an equivalent form:

(vii') if $P \Rightarrow Q$ and $R \subseteq AT$, then $P \cup R \Rightarrow Q \cup R$. (Augmentation law)

Example 2. In Table I, let $N=\{a,e\}$, then $\{a\} \subseteq N$, $\{e\} \subseteq N$, According to Theorem 14(i), we obtain $N \Rightarrow a$, $N \Rightarrow e$. In fact, $U_N(1)=U_N(2)=U_N(5) =\{1,2,4,5\}$,$U_N(3)=\{3,4,6,7\}$,$U_N(4)=\{1,2,\ldots,7\}$, $U_N(6) =U_N(7) =\{6,7\}$, So we know for $\forall x \in S$, $U_N(x) \subseteq U_a(x)$, $U_N(x) \subseteq U_e(x)$,therefore. $N \Rightarrow a$, $N \Rightarrow e$.

In traditional and complete information system rough set theory, the definition of knowledge dependency degree is given by using positive region. But in incomplete information system, equivalence relation does not valid again, so the equivalence class in equivalence relation has to be extended to tolerance class after equivalence relation is generalized to tolerance relation. That is, in incomplete information system, $POS_P(Q)$ has to be defined as

$$POS_P\ (Q)= \bigcup_{X \in U/R(Q)} \ \bigcup_{C \in U/R(P)} \{C \mid C \subseteq X\}. \qquad (25)$$

If the definition of knowledge dependency degree k in $P \xrightarrow{k} Q$ is still defined by $k=r_P(Q)= |POS_P(Q)|/|S|$, after analyzing that as long as for all attribute $a \in Q$ in attribute subset Q and for $\forall\ x \in S$, $f(x, a)=*$, $U_Q(x)=S$, it must have $POS_P(Q)=S$ at this time, then k=1. It is obtained that Q is completely dependent on P. Obvious that is not always true. For example, in Table I, for attribute set $P=\{c\}$ and attribute set $Q=\{a\}$, because $f(5,c)=f(6,c)=1$, and $f(5,a)=0$, $f(6,a)=1$, i.e. $f(5,a) \neq f(6,a)$, $Q=\{a\}$ is obviously not dependent on $P=\{c\}$. But according to the formula, it is obtained that $POS_P(Q)=S$, therefore $k=r_P(Q)=1$, Q is completely dependent on P. That is not the fact. So knowledge dependency degree in incomplete information system should be newly defined. In the following, we try to give a definition of knowledge dependency degree in two cases.

Case (i) : For attribute subsets P, $Q \subseteq AT$, any attribute in Q contains no null value *.

In this case, the definition of knowledge dependency degree k in $P \xrightarrow{k} Q$ can be given by using the extended dependency set.

Definition 10. For $\forall\ P, Q \subseteq AT$, the extended dependency set of Q on P is defined by

$$POS_P(Q)= \cup_{X \in S/IND(Q)} \cup_{C \in S/R(P)} \{C \mid C \subseteq X\},$$

where $IND(Q)$ is the indiscernbility relation or the equivalent relation determined by Q.

Especially, if $Q = \varnothing$, then $POS_P(Q)= \varnothing$.

At this time, the dependency set of Q on P is similar to that in traditional rough set theory, but it is determined by tolerance classes from tolerance relation.

Definition 11. Q is dependent on P by dependency degree, denoted by $P \xrightarrow{k} Q$, where

$$k=r_P\,(Q)= k(P,Q)=POS_P\,(Q)/|S|\ .$$

Obviously, $0 \le k \le 1$. When k=0, Q is not dependent on P; when k=1, Q is completely dependent on P, denoted by $P \Rightarrow Q$.

Case (ii) : For attribute subsets P, $Q \subseteq AT$, there exists at least one attribute in both P and Q contains null attribute value

*. At this time, new knowledge dependency degree is defined as follows.

Definition 12. Let P, $Q \subseteq AT$. For $\forall x, y \in S$, if $y \in U_P(x)$, then $y \in U_Q(x)$ must hold, and then Q is completely dependent on P, denoted by $P \Rightarrow Q$.

Definition 13. Let P, $Q \subseteq AT$ be any attribute subset in AT. $P \xrightarrow{k} Q$, meaning that Q is partially dependent on P by degree k, where

$$k=k(P,Q)=\sum_{x\in S}|U_P(x)\cap U_Q(x)|/\sum_{x\in S}|U_P(x)|. \tag{26}$$

It must have $0 \le k \le 1$. Because $x \in U_P(x) \neq \varnothing$, $x \in U_Q(x) \neq \varnothing$, $x \in U_P(x) \cap U_Q(x) \neq \varnothing$ for any P, $Q \subseteq AT$, thus, $0<k \le 1$; $k=0$ only if $Q = \varnothing$.

Obviously, if Q is completely dependent on P, i.e. $P \Rightarrow Q$, then it satisfies the condition in Definition 11. According to the computation of dependency degree defined in Definition 13, it must have $P \xrightarrow{k} Q$ in which $k=1$. So the definition in case (ii) is more general. In the following, we use the second definition to calculate dependency degree.

Theorem 16. $P \Rightarrow Q$ if, and only if $U_P(x) \subseteq U_Q(x)$ for $\forall x \in S$ and $\forall P, Q \subseteq AT$.

Proof. (i) If for $\forall x \in S, U_P(x) \subseteq U_Q(x)$, then it is obvious to obtain that $U_P(x) \cap U_Q(x)=U_P(x)$. Therefore, $k=1$ and $P \Rightarrow Q$.

(ii) If for $\forall P, Q \subseteq AT, P \Rightarrow Q$, then it is obtained that $k=1$, i.e.

$$\sum_{x\in S}|U_P(x)\cap U_Q(x)|/\sum_{x\in S}|U_P(x)|=1.$$

Thus, $\sum_{x\in S}|U_P(x)\cap U_Q(x)|=\sum_{x\in S}|U_P(x)|.$

For each x, $|U_P(x)\cap U_Q(x)|\le|U_P(x)|$. If we want $\sum_{x\in S}|U_P(x)\cap U_Q(x)|=\sum_{x\in S}|U_P(x)|$, it must have $U_P(x) \subseteq U_Q(x)$.

Synthesizing (i) and (ii), the theorem is true.

Example 3. In Table I, $U_a(1)=U_a(2)=U_a(5)=\{1,2,4,5\}$, $U_a(3)=U_a(6)=U_a(7)=\{3,4,6,7\}$, $U_a(4)=\{1,2,3,4,5,6,7\}$; $U_e(1)=U_e(2)=U_e(3)=U_e(4)=U_e(5)=U_e(6)=U_e(7)=\{1,2,3,4,5,6,7\}$. it is obtained that for $\forall x \in S$, $U_a(x) \subseteq U_e(x)$. So, $a \Rightarrow e$.

Example 4. Because $AT \Rightarrow e$, for any nonempty subset $B \subseteq AT$, for example, $B=\{a,b,c\}$, we have $B \Rightarrow e$. From the Definition 13, we obtain $a \xrightarrow{k_1} b$, $a \xrightarrow{k_2} c$, $c \xrightarrow{k_3} a$, $b \xrightarrow{k_4} c$, $c \xrightarrow{k_5} b$, where $k_1=29/31$, $k_2=25/31$, $k_3=29/43$, $k_4=39/45$, $k_5=39/43$.

V. Properties of KD and KD degree

Complete or partial knowledge dependency and knowledge dependency degree between attribute subsets in which an attribute may have null values are discussed.

Theorem 17. Let P, Q, $R \subseteq AT$. Then

(i) if $P \to Q$ and $Q \xrightarrow{k_1} R$, then $P \xrightarrow{k_2} R$ and $\alpha k_1 \ge k_2$, where $1 \le \alpha$ and $\alpha=\sum_{x\in S}|U_Q(x)|/\sum_{x\in S}|U_P(x)|$.

(ii) if $P \xrightarrow{k_1} Q$, $Q \Rightarrow R$, then $P \xrightarrow{k_2} R$, $k_1 \le k_2$.

Proof. (i) From $P \Rightarrow Q$, we obtain that $U_P(x) \subseteq U_Q(x)$ for $\forall x \in S$. Hence, $U_P(x) \cap U_R(x) \subseteq U_Q(x) \cap U_R(x)$. Thus $|U_P(x) \cap U_R(x)| \le |U_Q(x) \cap U_R(x)|$, and $|U_P(x)| \le |U_Q(x)|$.

$$\sum_{x\in S}|U_P(x)\cap U_R(x)| \le \sum_{x\in S}|U_Q(x)\cap U_R(x)|,$$

$$k_2=\sum_{x\in S}|U_P(x)\cap U_R(x)|/\sum_{x\in S}|U_P(x)|$$

$$\le \sum_{x\in S}|U_Q(x)\cap U_R(x)|/\sum_{x\in S}|U_P(x)|$$

$$=\sum_{x\in S}|U_Q(x)\cap U_R(x)|/\sum_{x\in S}|U_Q(x)|\cdot(\sum_{x\in S}|U_Q(x)|/\sum_{x\in S}|U_P(x)|)$$

$$=k_1\cdot(\sum_{x\in S}|U_Q(x)|/\sum_{x\in S}|U_P(x)|).$$

That is, $k_2 \le \alpha k_1$, $\alpha=\sum_{x\in S}|U_Q(x)|/\sum_{x\in S}|U_P(x)|$.

For $|U_P(x)| \le |U_Q(x)|$, we have $\sum_{x\in S}|U_P(x)|\le\sum_{x\in S}|U_Q(x)|$, and $1 \le \alpha$.

(ii) From $Q \Rightarrow R$, we obtain that $U_Q(x) \subseteq U_R(x)$ for $\forall x \in S$. Thus $U_P(x) \cap U_Q(x) \subseteq U_P(x) \cap U_R(x)$,

$|U_P(x) \cap U_Q(x)| \le |U_P(x) \cap U_R(x)|$,

$$\sum_{x\in S}|U_P(x)\cap U_Q(x)| \le \sum_{x\in S}|U_P(x)\cap U_R(x)|,$$

$$\sum_{x\in S}|U_P(x)\cap U_Q(x)|/\sum_{x\in S}|U_P(x)|$$

$$\le|\sum_{x\in S}|U_P(x)\cap U_R(x)|/\sum_{x\in S}|U_P(x)| .$$

So, $k_1 \le k_2$.

Example 5. (i) $a \to e$, $e \xrightarrow{k_1} a$, $b \xrightarrow{k_2} a$, where $k_1=31/49$, $k_2=29/45$, and

$$\alpha=\sum_{x\in S}|U_e(x)|/\sum_{x\in S}|U_a(x)|=49/31, \text{ and } 1 \le \alpha.$$

$\alpha k_1=(49/31)*(31/49)=1 \ge k_2=29/45$.

Thus, the correctness of Theorem 17(i) is checked.

(ii) In Table I, $a \xrightarrow{k_1} b$, $b \Rightarrow e$, $a \xrightarrow{k_2} e$, where $k_1=k_1=29/31$, $k_2=1$, $k_1 \le k_2$.

In Theorem 17, the transitivity of non-complete dependency of knowledge is studied and the dependency degrees before and after transferring are compared. The Theorem 17(i) shows that in this case the dependency degree after transferring is bigger than or equal to the dependency degree that before transferring, but the Theorem 17(ii) shows that it may not.

Theorem 18. Let P, Q, $R \subseteq AT$. Then

(i) if $P \xrightarrow{k_1} Q$ then $P \cup R \xrightarrow{k_2} Q$, $k_2 \le \alpha k_1$, where $1 \le \alpha$ and $\alpha = \sum_{x\in S} |U_P(x)| / \sum_{x\in S} |U_{P\cup R}(x)|$;

(ii) if $P \xrightarrow{k_1} Q$, $P \xrightarrow{k_2} Q \cup R$, then $k_1 \ge k_2$.

Proof. (i) $P \subseteq P \cup R$, so for $\forall x \in S$, $U_{P\cup R}(x) \subseteq U_P(x)$. Hence, $U_{P\cup R}(x) \cap U_Q(x) \subseteq U_P(x) \cap U_Q(x)$. $|U_{P\cup R}(x)| \le |U_P(x)|$, $|U_{P\cup R}(x) \cap U_Q(x)| \le |U_P(x) \cap U_Q(x)|$. Thus,

$$\sum_{x\in S} |U_{P\cup R}(x) \cap U_Q(x)| / \sum_{x\in S} |U_{P\cup R}(x)|$$
$$\le \sum_{x\in S} |U_P(x) \cap U_Q(x)| / \sum_{x\in S} |U_{P\cup R}(x)| .$$

Therefore, $k_2 \le \sum_{x\in S} |U_P(x) \cap U_Q(x)| / \sum_{x\in S} |U_P(x)|$

$$\cdot(\sum_{x\in S} |U_P(x)| / \sum_{x\in S} |U_{P\cup R}(x)|) = \alpha k_1 .$$

i.e. $k_2 \le \alpha k_1$, where $\alpha = \sum_{x\in S} |U_P(x)| / \sum_{x\in S} |U_{P\cup R}(x)|$. Since $U_{P\cup R}(x) \subseteq U_P(x), |U_{P\cup R}(x)| \le |U_P(x)|$, $\sum_{x\in S} |U_{P\cup R}(x)| \le \sum_{x\in S} |U_P(x)|$, thus $1 \le \alpha$.

(ii) Because $Q \subseteq Q \cup R$, for $\forall x \in S$, $U_{Q\cup R}(x) \subseteq U_Q(x)$. Thus $U_P(x) \cap U_{Q\cup R}(x) \subseteq U_P(x) \cap U_Q(x)$. Hence,

$|U_P(x) \cap U_{Q\cup R}(x)| \le |U_P(x) \cap U_Q(x)|$.

$$\sum_{x\in S} |U_P(x) \cap U_{Q\cup R}(x)| / \sum_{x\in S} |U_P(x)|$$
$$\le \sum_{x\in S} |U_P(x) \cap U_Q(x)| / \sum_{x\in S} |U_P(x)| .$$

Therefore, $k_1 \ge k_2$.

Example 6. (i)Let $P=\{a\}$,$R=\{b\}$,$S=P \cup R=\{a,b\}$. From the Table I, in $P \xrightarrow{k_1} c$,i.e. $a \xrightarrow{k_1} c$ and $P \cup R \xrightarrow{k_2} c$, i.e. $\{a,b\} \xrightarrow{k_2} c$, k_1=25/31, k_2=23/29, and $\sum_{x\in S} |U_P(x)| / \sum_{x\in S} |U_{P\cup R}(x)| = \sum_{x\in S} |U_a(x)| / \sum_{x\in S} |U_{\{a,b\}}(x)|$ =31 /29 ≥ 1. αk_1 =(31/29)*(25/31)=25/29 $\ge k_2$=23/29.

(ii) In Table I, let $P=\{a\}$,$Q=\{b\}$,$R=\{c\}$,$T=Q \cup R=\{b,c\}$. From Table I, it is obtained that $U_T(1)=U_T(3)=\{1,2,3,4,5,6\}$, $U_T(2)=U_T(5)=U_T(6)=$ {1,2,3,5,6,7}, $U_T(4)=\{1,3,4,7\}$, $U_T(7)=U_T(5)=\{2,4,5,6,7\}$,$P \xrightarrow{k_1} Q$, k_1=29/31. $P \xrightarrow{k_2} Q \cup R$, k_2=23/31. So $k_1 \ge k_2$.

So the correctness of (i),(ii) in the Theorem 18 is verified. The Theorem 18(i) shows that the dependency degree of the same knowledge to be dependent on much knowledge is still lesser than the original degree multiplying with a factor greeter than 1. The Theorem 18(ii) shows that for the same dependent knowledge, the much the knowledge, the lesser the dependent degree.

Theorem 19. Let P, Q, $R \subseteq AT$. If $Q \subseteq P$, $R \xrightarrow{k_1} P$, then $R \xrightarrow{k_2} Q$, and $k_1 \le k_2$.

Proof. For $Q \subseteq P$, $R(P) \subseteq R(Q)$,$P \Rightarrow Q$. Thus, for $\forall x \in S$,$U_P(x) \subseteq U_Q(x)$,$U_R(x) \cap U_P(x) \subseteq U_R(x) \cap U_Q(x)$, $|U_R(x) \cap U_P(x)| \le |U_R(x) \cap U_Q(x)|$,

$$\sum_{x\in S} |U_R(x) \cap U_P(x)| \le \sum_{x\in S} |U_R(x) \cap U_Q(x)| ,$$
$$\sum_{x\in S} |U_R(x) \cap U_P(x)| / \sum_{x\in S} |U_R(x)|$$
$$\le \sum_{x\in S} |U_R(x) \cap U_Q(x)| / \sum_{x\in S} |U_R(x)| .$$

That is, $k_1 \le k_2$.

Theorem 19 is another form of 18 (ii).

Example 7. In Table I, let $R=\{a\}$,$P=\{b, c\}$, $Q=\{b\}$, then $Q \subseteq P$, $R \xrightarrow{k_1} P=a \xrightarrow{k_1} \{b,c\}$, k_1=23/31; $R \xrightarrow{k_2} Q = a \xrightarrow{k_2} b$, k_2=29/31. So $k_1 \le k_2$.

The Theorem 19 expresses that depending on the same attribute set R, the degree of a subset Q of set P is always greater than the degree of set P.

Theorem 20. Let P, Q, $R \subseteq AT$. If $P \cup R \xrightarrow{k_1} Q$, then $P \xrightarrow{k_2} Q$, $R \xrightarrow{k_3} Q$, and min{ αk_2 , βk_3 } $\ge k_1$, where $1 \le \alpha$, and $\alpha = \sum_{x\in S} |U_P(x)| / \sum_{x\in S} |U_{P\cup R}(x)|$; $1 \le \beta$, and $\beta = \sum_{x\in S} |U_R(x)| / \sum_{x\in S} |U_{P\cup R}(x)|$.

Proof. Because $P \subseteq P \cup R$, $R \subseteq P \cup R$, thus $U_{P\cup R}(x) \subseteq U_P(x)$, $U_{P\cup R}(x) \subseteq U_R(x)$ for any $x \in S$. Hence,

$U_{P\cup R}(x) \cap U_Q(x) \subseteq U_P(x) \cap U_Q(x)$,

$U_{P\cup R}(x) \cap U_Q(x) \subseteq U_R(x) \cap U_Q(x)$,

$|U_{P\cup R}(x) \cap U_Q(x)| \le |U_P(x) \cap U_Q(x)|$,

$|U_{P\cup R}(x) \cap U_Q(x)| \le |U_R(x) \cap U_Q(x)|$,

$|U_{P\cup R}(x)| \le |U_P(x)|$,$|U_{P\cup R}(x)| \le |U_R(x)|$.

So

$$\sum_{x\in S} |U_{P\cup R}(x) \cap U_Q(x)| \le \sum_{x\in S} |U_P(x) \cap U_Q(x)| ,$$
$$\sum_{x\in S} |U_{P\cup R}(x) \cap U_Q(x)| \le \sum_{x\in S} |U_R(x) \cap U_Q(x)| ,$$
$$\sum_{x\in S} |U_{P\cup R}(x) \cap U_P(x)| / \sum_{x\in S} |U_{P\cup R}(x)|$$
$$\le \sum_{x\in S} |U_P(x) \cap U_Q(x)| / \sum_{x\in S} |U_{P\cup R}(x)|,$$
$$\sum_{x\in S} |U_{P\cup R}(x) \cap U_Q(x)| / \sum_{x\in S} |U_{P\cup R}(x)|$$
$$\le \sum_{x\in S} |U_R(x) \cap U_Q(x)| / \sum_{x\in S} |U_{P\cup R}(x)| ,$$

Thus, $\alpha k_2 \geq k_1$, $\alpha \geq 1$, $\alpha = \sum_{x\in S}|U_P(x)|/\sum_{x\in S}|U_{P\cup R}(x)|$;

$\beta k_3 \geq k_1, \beta \geq 1$, $\beta = \sum_{x\in S}|U_R(x)|/\sum_{x\in S}|U_{P\cup R}(x)|$.

Example 8. In Table 1, let $P=\{a\}$, $Q=\{b\}$, $R=\{c\}$, $P\cup R=\{a, c\}$, we obtain $P\cup R \xrightarrow{k_1} Q =\{a, c\} \xrightarrow{k_1} b$, where k_1=23/25. $P \xrightarrow{k_2} Q$, *i.e.* $a \xrightarrow{k_2} b$, $R \xrightarrow{k_3} Q$, *i.e.* $c \xrightarrow{k_3} b$, where k_2=29/31, k_3=39/43, $\alpha = \sum_{x\in S}|U_P(x)|/\sum_{x\in S}|U_{P\cup R}(x)|$ =31/25, and $1 \leq \alpha$, $\beta = \sum_{x\in S}|U_R(x)|/\sum_{x\in S}|U_{P\cup R}(x)|$ =43/25, and $1 \leq \beta$. αk_2 =(31/25)*(29/31) =29/25, βk_3=(43/25)*(39/43) =39/25. min{29/25,39/25} $\geq$ 23/25 = k_1. So min{ $\alpha k_2, \beta k_3$ } $\geq k_1$.

Therefore, it verifies the correctness of the Theorem 20. Theorem 20 reveals some regularities of knowledge dependency degree on partial knowledge and the entire knowledge.

In addition, the following result can be obtained.

Theorem 21. Let P, Q, $R \subseteq AT$. If $P \Rightarrow Q$, $R \xrightarrow{k_1} Q$, then $R \xrightarrow{k_2} P$, and $k_1 \geq k_2$.

Proof. From $P \Rightarrow Q$, it can obtained that for $\forall\ x \in S$, $U_P(x) \subseteq U_Q(x)$. Furthermore, $U_R(x) \cap U_P(x) \subseteq U_R(x) \cap U_Q(x)$. Thus,

$|U_R(x) \cap U_P(x)| \leq |U_R(x) \cap U_Q(x)|$,

$$\sum_{x\in S}|U_R(x) \cap U_P(x)| \leq \sum_{x\in S}|U_R(x) \cap U_Q(x)|,$$

$$\sum_{x\in S}|U_R(x) \cap U_P(x)|/\sum_{x\in S}|U_R(x)|$$

$$\leq \sum_{x\in S}|U_R(x) \cap U_Q(x)|/\sum_{x\in S}|U_R(x)|.$$

Therefore, $k_1 \geq k_2$.

Theorem 22. Let P, Q, $R \subseteq AT$. If $P \Rightarrow Q$, $P \xrightarrow{k_1} R$, then $Q \xrightarrow{k_2} R$ and $\alpha k_2 \geq k_1$ where $\alpha \geq 1$ and $\alpha = \sum_{x\in S}|U_Q(x)|/\sum_{x\in S}|U_P(x)|$.

Proof. From $P \Rightarrow Q$, it is obtained that for $\forall\ x \in S$, $U_P(x) \subseteq U_Q(x)$. Furthermore, $U_R(x) \cap U_P(x) \subseteq U_R(x) \cap U_Q(x)$, i.e. $U_P(x) \cap U_R(x) \subseteq U_Q(x) \cap U_R(x)$. Therefore,

$|U_P(x) \cap U_R(x)| \leq |U_Q(x) \cap U_R(x)|$,

$|U_P(x)| \leq |U_Q(x)|$,

$$\sum_{x\in S}|U_P(x) \cap U_R(x)| \leq \sum_{x\in S}|U_Q(x) \cap U_R(x)|,$$

$$\sum_{x\in S}|U_P(x) \cap U_R(x)|/\sum_{x\in S}|U_P(x)|$$

$$\leq \sum_{x\in S}|U_Q(x) \cap U_R(x)|/\sum_{x\in S}|U_P(x)|.$$

Thus, $k_2\sum_{x\in S}|U_Q(x)|/\sum_{x\in S}|U_P(x)| = \alpha k_2 \geq k_1$, $\alpha \geq 1$, and $\alpha = \sum_{x\in S}|U_Q(x)|/\sum_{x\in S}|U_P(x)|$.

VI. A REDUCTION ALGORITHM FOR AN IDT

For an incomplete decision table $IDT=(U,AT,V,f)$, where $AT=C \cup \{d\}$, C is the set of condition attributes. d is the decision attribute. From $R(AT\text{-}\{a\}) \subseteq R(AT)$, i.e., $AT\text{-}\{a\} \rightarrow AT$, we know that, a is dispensable attribute in AT, and because $k(AT\text{-}\{a\},AT)$=1 is equivalent to $AT\text{-}\{a\} \rightarrow AT$, so for the incomplete decision table, if $k((C\text{-}\{a\}) \cup \{d\}, C \cup \{d\})$=1, then a is a dispensable attribute. According to this law and the above defined attribute dependence degree, we can design a new attribute reduction algorithm for an incomplete decision table as follows.

Algorithm A reduction algorithm for an incomplete decision table

Step 1. Set $R \Leftarrow C$.

Step 2. For each $a \in C$, compute $k((R\text{-}\{a\}) \cup \{d\}, R \cup \{d\})$.

Step 3. Find $k((R\text{-}\{a\}) \cup \{d\}, R \cup \{d\}) = max\{k((R\text{-}\{b\}) \cup \{d\}, R \cup \{d\}) | b \in A\}$. If $k((R\text{-}\{a\}) \cup \{d\}, R \cup \{d\})$=1, then a is dispensable, and set $R \Leftarrow R\text{-}\{a\}$. If there exist many attributes satisfying $k(R\text{-}\{a\}, R \cup \{d\})$=1. Go to Step 2.

Step 4. R is a reduction of A. Output R, and end the algorithm.

VII. AN APPLICATION OF KD TO ROBOT ROD CATCHING CONTROL

A robot rod catching control incomplete decision table has been shown in Table I. 6 state attributes describing situations for the robot are: a, b, c, d, e, f. $C=\{a, b, c, d, e\}$ is the condition attribute set. f is a decision attribute.

Table II. The reduct of Table I

	a	b	c	d	f
1	0	0	*	*	1
2	0	*	1	0	1
3	1	0	*	*	1
4	*	*	0	1	2
5	0	*	1	*	2
6	1	*	1	0	3
7	1	1	*	*	4

a=1 means the rod locates at the central point. a=0 means not. b=1 states the robot faces to the rod, b=0 means not. c=1 denotes the robot is at the central line of the rod, c=0 means not. d=1 means the robot is facing forward to the central line of the rod. d=0 means not. e=1 denotes the robot grasps. e=0 means not. f=1 represents the behavior of the robot is rotating. f=2 means moving forward. f=3 means catching, f=4 means stopping. * in the table means non-determine. So it is an incomplete decision table.

Using knowledge dependency, dependency degree and the algorithm in the above, we let R<=C, and obtain $k(R-\{e\}\cup\{f\},R\cup\{f\}=1$. That means *e* is dispensable in *AT*. So *R*<=*R*-{*e*}, i.e. *e* is removed from condition attribute set *C*. In this way, we get that *R*= {*a*, *b*, *c*, *d*} is the unique reduct of the decision table, see Table II. It means the behavior of the robot is dominated by attribute *a*, *b*, *c*, *d*.

VIII. Conclusion

Some relationships between tolerance classes and dispensable attribute in incomplete information system are revealed in the present paper. Through defining dependency especially partial dependency and partial dependency degree, the properties of dispensable attribute in attribute set and dependency laws are explored in tolerance rough set model. Several necessary and/or sufficient conditions for dispensable attribute are obtained. Complete knowledge dependency possessing reflexivity, transitivity, augmentation, and decomposition laws in incomplete information system are given. Partial knowledge dependency and partial knowledge dependency degree including complete knowledge dependency satisfying special laws after transferring, augmentation, and decomposition are proved with several theorems. Based on knowledge dependency and knowledge dependency degree, an algorithm to solve the reduct of an incomplete decision table is designed. Applied concepts, algorithms, theorem results discussed and suggested in the paper to solve the reduct of robot rod catching problem, it reaches at a good result. This verifies the correctness of what we have done.

References

[1] Z. Pawlak, "Rough set theory and its applications to data analysis," *J. Cybernetics and Systems*. Vol.29, No.7, 1998, pp.661-688.
[2] Z. Pawlak, "Rough sets and intelligent data analysis," *Information Sciences*. Vol.147,No.1-4,2002, pp.1-12
[3] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*, Dordecht: Kluwer Academic Publishers.1991.
[4] M. Kryszkiewicz, "Rough set approach to incomplete information systems," *Information Sciences*. Vol.112, 1998, pp.39-49.
[5] J. Stefanowski, "Incomplete information tables and rough classification," *J. Computational Intelligence*, Vol.17, No.3,2001, pp.545-566.
[6] J. Stefanowski, A. Tsoukias, "On the extension of rough sets under incomplete information," *Proc of the 7th Int'l Workshop on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing*, Berlin: Springer-Verlag. 1999, pp.73-81.
[7] G. Y. Wang, "Extension of rough set under incomplete information systems," *J. Computer Research and Development*. Vol.39, No. 10, 2002, pp.1238-1243.
[8] J. Y. Leung, D. Y. Li, "Maximal consistent block technique for rule acquisition in incomplete information systems," *Information Sciences*, Vol.153,2003, pp.85-106.
[9] J. Y. Leung, Z. Z. Shi, "The information entropy, rough entropy and knowledge granulation in rough set theory," *Int. J. Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol.12,No.1, 2004,pp.37-46
[10] C. Wu, X. B. Yang, "Information granules in general and complete covering," *Proc. of the 2005 IEEE International Conference on Granular Computing*. Vol.2,2005, pp.675-678.
[11] W. L. Chen, J. X. Cheng, C. J. Zhang, "A generalized model of rough set theory based on compatibility relation," *J. computer engineering and applications*, Vol.16, No. 4, 2004, pp.26-28 .
[12] J. S. Mi, W. Z. Wu, W. X. Zhang, "Approaches to knowledge reduction based on variable precision rough set model," *Information Sciences*. Vol.159, No. 3-4,2004, pp.255-272.
[13] C. Wu, X. H. Hu, Z. Li, X. H. Zhou, P. Achananuparp, "Algorithms for different approximations in incomplete information systems with maximal compatible classes as primitive granules," *Proc. of IEEE Int. Conf. on Granular Computing*, GrC 2007, San Jose, California, USA.2007, pp.169-174.
[14] Y. H. Qian, J. Y. Liang, C. Y. Dang, "Incomplete multigranulation rough set," *IEEE Transactions on Systems, Man and Cybernetics*(Part A).Vol.40,No. 6, 2010, pp. 420-431.
[15] W. H. Xu, G. H. Zhang, W. X. Zhang, "Lower-approximation distribution reduct and rules in incomplete information system," *J. Xi'an Institute of Technology*,Vol.24, No. 4, 2004, pp. 386-390.
[16] W. X. Zhang, J. Y. Liang, *Rough Set Theory and Approaches*, Science Press,2001.
[17] Xiaofei Deng, Yiyu Yao, "Decision-theoretic three-way approximations of fuzzy sets," Information Sciences, Vol.279, 2014, pp.702–715.
[18] D. Hu, H. X. Li, "The measurement of dependent degree of knowledge," *J. Beijing Normal University*,Vol.40,No.3,2004, pp.320-325.

Chen Wu was born in 1962 in Hubei province in China. He got his Ph.D from Nanjing university of science and technology in 2007 in Nanjing city in Jiangsu province in China, majoring in computer science and engineering.

He was a visiting professor at information school at Drexel university in 2006 in Philadelphia in USA. Current and previous research interests for Professor and Ph.D. Chen Wu are: rough set, data mining, pattern recognition and data processing and algorithm design.

Wei Zhu is a post graduate students and are now studying in the school of computer science and engineering at the Jiangsu university of science and technology. Research areas are: rough set, data mining, decision making.

Lijuan Wang was born in 1981 in Jiangsu province in China. She got her Ph.D. from Nanjing university of science and technology in 2014 in Nanjing city in Jiangsu province in China, majoring in computer science and engineering. She is an assistant professor. Her current research areas are: rough set, data mining.