Available online at www.sciencedirect.com

ScienceDirect

Computers and Mathematics with Applications 55 (2008) 1754–1765

An International Journal computers & mathematics with applications

www.elsevier.com/locate/camwa

Converse approximation and rule extraction from decision tables in rough set theory

Yuhua Qian[a,b,c], Jiye Liang[a,b,*], Chuangyin Dang[c]

[a] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Taiyuan, 030006, China

[b] School of Computer and Information Technology, Shanxi University, Taiyuan, 030006, China

[c] Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong

Received 14 November 2006; received in revised form 14 August 2007; accepted 30 August 2007

Abstract

In this paper, the concept of a granulation order is proposed in an information system. The converse approximation of a target concept under a granulation order is defined and some of its important properties are obtained, which can be used to characterize the structure of a set approximation. For a subset of the universe in an information system, its converge degree is monotonously increasing under a granulation order. This means that a proper family of granulations can be chosen for a target concept approximation according to user requirements. As an application of the converse approximation, an algorithm based on the converse approximation called REBCA is designed for decision-rule extraction from a decision table, which has a time complexity of $O(\frac{m}{2}|C|^2|U|\log_2|U|)$, and its practical applications are illustrated by two examples.

Keywords: Rough set theory; Converse approximation; Decision table; Decision rules; Rule extracting

1. Introduction

Recently, rough set theory developed by Pawlak in [1] has become a popular mathematical framework for the analysis of vague descriptions of objects. The focus of rough set theory is on the ambiguity caused by limited discernibility of objects in the domain of discourse. Its key concepts are those of object indiscernibility and set approximation, and its main perspectives are information view and algebra view [2]. The primary use of rough set theory has so far mainly been in generating logical rules for classification and prediction using information granules; thereby making it a prospective tool for pattern recognition, image processing, feature selection, data mining and knowledge discovery process from large data sets [3–5].

As a recently renewed research topic, granular computing (GrC) is an umbrella term to cover any theories, methodologies, techniques, and tools that make use of granules in problem solving [6–8]. Basic ideas of GrC have appeared in related fields, such as interval analysis, rough set theory, cluster analysis, machine learning,

* Corresponding author at: Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Taiyuan, 030006, China.

E-mail addresses: jinchengqyh@126.com (Y. Qian), ljy@sxu.edu.cn (J. Liang), mecdang@cityu.edu.hk (C. Dang).

0898-1221/$ - see front matter
doi:10.1016/j.camwa.2007.08.031

databases, and many other. As follows, for our further development, we briefly review research on GrC. L.A. Zadeh identified three basic concepts that underlie the process of human cognition, namely, granulation, organization, and causation. “Granulation involves decomposition of whole into parts, organization involves integration of parts into whole, and causation involves association of causes and effects”. A general framework of granular computing was presented by Zadeh [6] in the context of fuzzy set theory. Some authors [9–11] examined granular computing in connection with the theory of rough sets. Yao [12] suggested the use of hierarchical granulations for the study of stratified rough set approximations. Lin [13] and Yao [14] studied granular computing using neighborhood systems. Klir [15] investigated some basic issues of computing with granular computing with granular probabilities. The theory of quotient space had been extended into the theory of fuzzy quotient space based on fuzzy equivalence relation [16]. Liang and Shi [17] established the relationship among knowledge granulation, information entropy, granularity measure and rough entropy in rough set theory. Liang and Qian [18,19] studied rough sets approximation based on dynamic granulation (positive approximation) and its application for rule extracting. In the view of granular computing, a general concept described by a set is always characterized via the so-called upper and lower approximations under static granulation in rough set theory, and a static boundary region of the concept is induced by the upper and lower approximations. However, in rough sets approximation under dynamic granulation, a general concept described by using the positive approximation is characterized via variational upper and lower approximations under dynamic granulation, which is an aspect of people’s comprehensive solving ability at some different granulations.

In recent years, rough set theory has been widely applied to extracted from decision tables. Decision tables have two cases: consistent decision tables and inconsistent decision tables. Ziarko [20] proposed a method of rule extracting from decision tables by reducing boundary area in decision tables. There are at least three approaches to reducing boundary area. The first and simplest technique is to try to increase the decision table “resolution” by adding more attributes or by increasing the precision of existing ones. The second approach is to provide another layer of decision tables, by treating each subdomain of objects matching the description of an elementary set of the boundary area of the original decision tables as a domain (the universe) by itself. The third proposed method of boundary area reduction is based on the idea of treating the subdomain of the original domain corresponding to the whole boundary area as the new domain by itself. However, in fact, the classification accuracy (the approximation measure) [21] is constrained according to decision requirements or preference of decision makers in general. An obvious question is how to extract much simpler decision rules on the basis of keeping an approximation measure. Liang and Qian [18,19] presented the notion of positive approximation and applied it for rule extracting from consistent decision tables in rough set theory. In [21], a given relative knowledge reduction determines a family of decision rules for a decision table. That is to say, relative knowledge reduction must be obtained before rule extracting from decision tables. Many types of knowledge reductions have been proposed [22–33] in the area of rough set and each of the reductions aimed at some basic requirement. However, the complexity of these attribute reductions are much worse, which is inconvenient to extract decision rules from decision tables. Our research aims to find a method for rule extracting without computing relative attribute reduction of a decision table in rough set theory. Based on these studies, the main objectives of this paper are to establish the structure of the approximation of a target concept by introducing a notion of a granulation order, investigate some of its important properties, and apply it to rule extracting from decision tables.

The rest of this paper is organized as follows. In Section 2, we review some basic concepts and properties of the positive approximation. The definitions of converse approximations of a target concept and a target decision (target partition) based on dynamic granulation are presented respectively, and some of their very useful properties are educed in Section 3. In Section 4, a new rule-extracting method of decision tables based on converse approximation in rough set theory is proposed and time complexity of this algorithm is analysed. And we show how the algorithm to extract decision rules by two illustrative examples (a consistent decision table and an inconsistent decision table). Finally, Section 5 concludes the whole paper.

2. Positive approximation

In this section we briefly introduce some existing basic concepts and properties of the positive approximation that are relative to the research in this paper. These concepts and properties will be helpful for us to understand the notion of a granulation order and set approximation under dynamic granulation and to establish converse approximation in this paper.

Let $S = (U, A)$ be an information system, where U is a non-empty, finite set of objects and A is a non-empty finite set of attributes. Let $P, Q \in 2^A$ be two attribute subsets, where 2^A is a power set of A. By $\mathrm{IND}(P)$ and $\mathrm{IND}(Q)$, we denote the indiscernible relations induced by P and Q, respectively. A partial relation $\preceq$ on 2^A is defined in [25] as follows: $P \preceq Q$ ($Q \succeq P$) if and only if, for every $P_i \in U/\mathrm{IND}(P)$, there exists $Q_j \in U/\mathrm{IND}(Q)$ such that $P_i \subseteq Q_j$, where $U/\mathrm{IND}(P) = \{P_1, P_2, \ldots, P_m\}$ and $U/\mathrm{IND}(Q) = \{Q_1, Q_2, \ldots, Q_n\}$ are partitions induced by $\mathrm{IND}(P)$ and $\mathrm{IND}(Q)$, respectively. A partition induced by an equivalence relation provides a granulation world for describing a target concept. Thus, a sequence of granulation worlds from coarse to fine can be determined by a sequence of attribute sets with granulations from coarse to fine in the power set of A. The positive approximation gives the definition of the upper and lower approximations of a target concept under a given granulation order [18]. It can be used to extract from a given decision table decision rules with granulations from coarse to fine. The definition of the positive approximation is as follows.

Definition 2.1 (*[18]*). Let $S = (U, A)$ be an information system, $X \subseteq U$ and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \succeq R_2 \succeq \cdots \succeq R_n$ ($R_i \in 2^A$). Let $P_i = \{R_1, R_2, \ldots, R_i\}$, we define P_i-upper approximation $\overline{P_i}(X)$ and P_i-lower approximation $\underline{P_i}(X)$ of P_i-positive approximation of X as

$$\overline{P_i}(X) = \overline{R_i}(X),$$

$$\underline{P_i}(X) = \bigcup_{k=1}^{i} \underline{R_k} X_k,$$

where $X_1 = X$ and $X_k = X - \bigcup_{j=1}^{k-1} \underline{R_j} X_j$ for $k = 2, 3, \ldots, n$, $i = 1, 2, \ldots, n$.

Theorem 2.1 (*[18]*). *Let $S = (U, A)$ be an information system, $X \subseteq U$ and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \succeq R_2 \succeq \cdots \succeq R_n$ ($R_i \in 2^A$). Let $P_i = \{R_1, R_2, \ldots, R_i\}$, then $\forall P_i$ ($i = 1, 2, \ldots, n$), we have*

$$\underline{P_i}(X) \subseteq X \subseteq \overline{P_i}(X),$$

$$\underline{P_1}(X) \subseteq \underline{P_2}(X) \subseteq \cdots \subseteq \underline{P_n}(X).$$

Theorem 2.1 states that the lower approximation of the positive approximation of a target concept enlarges as a granulation order becomes longer through adding an equivalence relation, which will help to exactly describe the target concept.

Theorem 2.2 (*[18]*). *Let $S = (U, A)$ be an information system, $X \subseteq U$ and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \succeq R_2 \succeq \cdots \succeq R_n$ ($R_i \in 2^A$). Let $P_i = \{R_1, R_2, \ldots, R_i\}$, then $\forall P_i$ ($i = 1, 2, \ldots, n$), we have*

$$\alpha_{P_1}(X) \leq \alpha_{P_2}(X) \leq \cdots \leq \alpha_{P_n}(X),$$

where $\alpha_{P_i}(X) = \frac{|\underline{P_i}(X)|}{|\overline{P_i}(X)|}$ is the approximation measure of X with respect to P.

The approximation measure was introduced to the positive approximation in order to describe the uncertainty of a target concept under a granulation order [18]. From this theorem, one can see that the approximation measure of a target concept enlarges as a granulation order becomes longer through adding an equivalence relation.

The main motivation of the positive approximation is to extend rough set approximation under static granulation to rough set approximation under dynamic granulation and to approach a target concept by the change in a granulation order. Through the positive approximation, one can extract from a given decision table a family of decision rules with granulations from coarse to fine. In some practical issues, however, we need to mining decision rules on the basis of keeping the approximation measure of a target concept. Obviously, the positive approximation appears not to be suited for rule extracting from decision tables on the basis of keeping the approximation measure of every decision class in decision partition on the universe. Therefore, in the next section, we introduce a notion of converse approximation in rough set theory.

3. Converse approximation

In this section, we introduce a new set-approximation approach called a converse approximation and investigate some of its important properties.

Let $S = (U, A)$ be an information system and $R \in 2^A$. A partition induced by the equivalence relation $\mathrm{IND}(R)$, provides a granulation world for describing a target concept X. So a sequence of attribute sets $R_i \in 2^A$ $(i = 1, 2, \ldots, n)$ with $R_1 \preceq R_2 \preceq \cdots \preceq R_n$ can determine a sequence of granulation worlds from fine to coarse. A converse approximation can give the definition of the upper and lower approximations of a target concept under a granulation order. In the following, we introduce the definition of the converse approximation of a target set under dynamic granulation.

Definition 3.1. Let $S = (U, A)$ be an information system, $X \subseteq U$ and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \preceq R_2 \preceq \cdots \preceq R_n$ $(R_i \in 2^A)$. Let $P_i = \{R_1, R_2, \ldots, R_i\}$, we define P_i-upper approximation $\overline{P_i}(X)$ and P_i-lower approximation $\underline{P_i}(X)$ of P_i-converse approximation of X as

$$\overline{P_i}(X) = \overline{R_1}(X),$$

$$\underline{P_i}(X) = \underline{R_i}X_i \cup \left(\bigcup_{k=1}^{i-1}(\underline{R_k}X_k - \underline{R_{k+1}}X_k)\right),$$

where $X_1 = X$ and $X_{k+1} = \underline{R_k}X_k$ for $k = 1, 2, \ldots, n$, $i = 1, 2, \ldots, n$.

Definition 3.1 shows that a target concept can be approached by the change in the lower approximation $\underline{P_i}(X)$ and the upper approximation $\overline{P_i}(X)$. In particular, we call $\overline{P_n}(X) = \overline{R_1}(X)$ and $\underline{P_n}(X) = \underline{R_n}X_n \cup (\bigcup_{k=1}^{n-1}(\underline{R_k}X_k - \underline{R_{k+1}}X_k))$ P-upper approximation and P-lower approximation of P-converse approximation of X, respectively.

Theorem 3.1. *Let $S = (U, A)$ be an information system, $X \subseteq U$ and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \preceq R_2 \preceq \cdots \preceq R_n$ $(R_i \in 2^A)$. Let $P_i = \{R_1, R_2, \ldots, R_i\}$, then $\forall P_i$ $(i = 1, 2, \ldots, n)$, we have that*

$$\underline{P_i}(X) \subseteq X \subseteq \overline{P_i}(X),$$

$$\underline{P_1}(X) = \underline{P_2}(X) = \cdots = \underline{P_n}(X).$$

Proof. If follows from Definition 3.1 that $\underline{R_i}X_i \subseteq X_i$ and $X_{i+1} \subseteq X_i$ hold for $\underline{R_i}X_i$. Therefore, one can obtain that

$$\begin{aligned}\underline{P_i}(X) &= \underline{R_i}X_i \cup \left(\bigcup_{k=1}^{i-1}(\underline{R_k}X_k - \underline{R_{k+1}}X_k)\right)\\ &= \underline{R_i}X_i \cup \underline{R_1}X_1 = \underline{R_1}X_1 = \underline{R_1}X,\end{aligned}$$

i.e., $\underline{P_1}(X) = \underline{P_2}(X) = \cdots = \underline{P_n}(X) = \underline{R_1}(X)$.

Moreover, it is clear from Definition 3.1 that

$$\underline{P_i}(X) = \underline{R_1}(X) \subseteq X \subseteq \overline{P_1}(X) = \overline{P_i}(X).$$

Thus $\underline{P_i}(X) \subseteq X \subseteq \overline{P_i}(X)$. This completes the proof. □

Theorem 3.1 states that the lower approximation and the upper approximation of P-converse approximation of a target concept do not change as a granulation order becomes longer through adding equivalence relations, which gives a new method of describing a target concept. In particular, the number of equivalence classes in $\underline{P_i}(X)$ decreases as the granulation order become longer. In other words, some new equivalence classes under different granulations are induced by combining some known equivalence classes in the lower approximation of the target concept.

In order to illustrate the essence that the converse approximation is concentrated on the changes in the construction of the target concept X (equivalence classes in lower approximation of X with respect to P), we can redefine P-converse approximation of X by using some equivalence classes on U. Therefore, the structure of P-upper approximation $\overline{P}(X)$ and P-lower approximation $\underline{P}(X)$ of P-converse approximation of X are represented as follows:

$$[\overline{P}(X)] = \{[x]_{R_1} \mid [x]_{R_1} \cap X \neq \emptyset\},$$

$$[\underline{P}(X)] = \left\{\bigcup_{i=1}^{n}[x]_{R_i} \mid [x]_{R_i} \subseteq \underline{R_i}X_i \cup \left(\bigcup_{k=1}^{i-1}(\underline{R_k}X_k - \underline{R_{k+1}}X_k)\right)\right\},$$

where $X_1 = X$, $X_{k+1} = \underline{R_k} X_k$ for $i = 1, 2, \ldots, n$ and $[x]_{R_i}$ represents the equivalence class obtaining x in the partition U/R_i. It follows from the structure of $[\underline{P}(X)]$ that

$$\underline{R_k} X_k - \underline{R_{k+1}} X_k = \{[x]_{R_k} \mid [x]_{R_k} \subseteq \underline{R_k} X_k \cap bn_{R_{k+1}}(X_k),\ x \in U\},$$

where $bn_{R_{k+1}}(X_k) = \overline{R_{k+1}}(X_k) - \underline{R_{k+1}}(X_k)$. That is to say, these objects in $\underline{R_k} X_k - \underline{R_{k+1}} X_k$ will result in a new boundary region if they are described by R_{k+1}. We only research new lower approximation induced by dynamic granulation order every time for describing the target concept.

Example 3.1. Let $U = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$, $X = \{e_1, e_2, e_3, e_4, e_7, e_8\}$ and $U/R_1 = \{\{e_1\}, \{e_2\}, \{e_3, e_4\}, \{e_5, e_6\}, \{e_7, e_8\}\}$, $U/R_2 = \{\{e_1\}, \{e_2\}, \{e_5, e_6\}, \{e_3, e_4, e_7, e_8\}\}$ be two partitions on U.

Obviously, $R_1 \preceq R_2$ holds. Thus one can construct two granulation orders (a family of equivalence relations) $P_1 = \{R_1\}$ and $P_2 = \{R_1, R_2\}$.

By computing the converse approximation of X with respect to P, one can easily obtain that

$$[\overline{P_1}(X)] = \{\{e_1\}, \{e_2\}, \{e_3, e_4\}, \{e_7, e_8\}\},$$
$$[\underline{P_1}(X)] = \{\{e_1\}, \{e_2\}, \{e_3, e_4\}, \{e_7, e_8\}\},$$
$$[\overline{P_2}(X)] = \{\{e_1\}, \{e_2\}, \{e_3, e_4\}, \{e_7, e_8\}\}$$

and

$$[\underline{P_2}(X)] = \{\{e_1\}, \{e_2\}, \{e_3, e_4, e_7, e_8\}\},$$

where $\{e_1\}$, $\{e_2\}$ can be induced by both equivalence relation R_1 and equivalence relation R_2, $\{e_3, e_4, e_7, e_8\}$ is only obtained by equivalence relation R_2. That is to say, the target concept X is described by using granulation $P_1 = \{R_1\}$ and $P_2 = \{R_1, R_2\}$. □

In order to discuss the properties of the converse approximation based on dynamic granulation, we need to introduce the definition of $\sqsubseteq$. Assume A, B be two families of classical sets, where $A = \{A_1, A_2, \ldots, A_m\}$, $B = \{B_1, B_2, \ldots, B_n\}$. We say $A \sqsubseteq B$ if and only if, for any $A_i \in A$, there exists $B_j \in B$ such that $A_i \subseteq B_j$ $(i \le m,\ j \le n)$. From this denotation, we can obtain the following theorem.

Theorem 3.2. *Let $S = (U, A)$ be an information system, $X \subseteq U$ and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \preceq R_2 \preceq \cdots \preceq R_n$ $(R_i \in 2^A)$. Let $P_i = \{R_1, R_2, \ldots, R_i\}$, then $\forall P_i$ $(i = 1, 2, \ldots, n)$, we have*

$$[P_1(X)] \sqsubseteq [P_2(X)] \sqsubseteq \cdots \sqsubseteq [P_n(X)].$$

Proof. It follows from Definition 3.1 and Theorem 3.1 that

$$\underline{P_1}(X) = \underline{P_2}(X) = \cdots = \underline{P_n}(X) = \underline{R_1}(X).$$

Suppose $1 \le i < j \le n$, then $R_i \preceq R_j$. Therefore, one can get that

$$\begin{aligned}
[\underline{P_i}(X)] &= \underline{R_i} X_i \cup \left(\bigcup_{k=1}^{i-1} (\underline{R_k} X_k - \underline{R_{k+1}} X_k) \right) \\
&\sqsubseteq \left(\bigcup_{k=1}^{i-1} (\underline{R_k} X_k - \underline{R_{k+1}} X_k) \right) \cup \left(\bigcup_{k=i}^{j-1} (\underline{R_k} X_k - \underline{R_{k+1}} X_k) \right) \cup \underline{R_j} X_j \\
&= \underline{R_j} X_j \cup \left(\bigcup_{k=1}^{j-1} (\underline{R_k} X_k - \underline{R_{k+1}} X_k) \right) \\
&= [\underline{P_j}(X)].
\end{aligned}$$

Therefore $[P_1(X)] \sqsubseteq [P_2(X)] \sqsubseteq \cdots \sqsubseteq [P_n(X)]$. This completes the proof. □

Theorem 3.2 states that the number of classes in the lower approximation of P-converse approximation of a target concept decreases as the granulation order become longer through adding equivalence relations, and some new equivalence classes are induced by combining known equivalence classes in the lower approximation of the target concept. This mechanism can reduce the number of equivalence classes for describing a target concept on the basis of keeping the approximation measure $\alpha_P(X) = \frac{|\underline{P}(X)|}{|\overline{P}(X)|}$, which can simply the construction of the approximation of a target concept. In fact, through the converse approximation, we can more clearly understand the rough approximation of a target concept.

Let $S = (U, A)$ be an information system, $R \in 2^A$ a subset of attributes on A and $\Gamma = \{X_1, X_2, \ldots, X_m\}$ a partition on U. Lower approximation and upper approximation of Γ with respect to R are defined by

$$\underline{R}\Gamma = \{\underline{R}X_1, \underline{R}X_2, \ldots, \underline{R}X_m\},$$
$$\overline{R}\Gamma = \{\overline{R}X_1, \overline{R}X_2, \ldots, \overline{R}X_m\}.$$

In the view of describing the target concept by using equivalence classes, we need to define a new measure to discuss the converge situation about equivalence classes in a lower approximation, which will be helpful in understanding the construction of the target concept in rough set theory. For this objective, we introduce the following notion called a converge degree.

Definition 3.2. Let $S = (U, A)$ be an information system, $R \in 2^A$ a subset of attributes on A and $\Gamma = \{X_1, X_2, \ldots, X_m\}$ a partition on U. Converge degree of Γ with respect to R as

$$c(R, \Gamma) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \sum_{j=1}^{s_i} p^2(X_i^j),$$

where s_i is the number of equivalence classes in $\underline{R}X_i$, $p(X_i^j) = \frac{|X_i^j|}{|X_i|}$ and X_i^j $(1 \le j \le s_i)$ is an equivalence class in $\underline{R}X_i$.

Obviously, $0 \le c(R, \Gamma) \le 1$. In particular,

(1) if $\Gamma = \{X\}$, then $c(R, \Gamma) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \sum_{j=1}^{s_i} p^2(X_i^j) = \sum_{j=1}^{s_i} p^2(X_i^j)$, i.e., the converge degree of Γ with respect to R degenerates into the converge degree of X with respect to R, denoted by $c(R, X)$;

(2) if $U/R = \omega = \{\{x\} \mid x \in U\}$, then $c(R, \Gamma) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \times \frac{|X_i|}{|X_i|^2} = \frac{m}{|U|}$;

(3) if $U/R = \delta = \{U\}$, then $c(R, \Gamma)$ achieves its minimum value $c(R, \Gamma) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \times 0 = 0$; and

(4) if $U/R = \Gamma$, then $c(R, \Gamma)$ achieves its maximum value $c(R, \Gamma) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \times 1 = 1$.

Let $S = (U, A)$ be an information system, $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \preceq R_2 \preceq \cdots \preceq R_n$ $(R_i \in 2^A)$ and $\Gamma = \{X_1, X_2, \ldots, X_m\}$ be a partition on U. Lower approximation and upper approximation of Γ with respect to P are defined by

$$\underline{P}\Gamma = \{\underline{P}X_1, \underline{P}X_2, \ldots, \underline{P}X_m\},$$
$$\overline{P}\Gamma = \{\overline{P}X_1, \overline{P}X_2, \ldots, \overline{P}X_m\}.$$

Similar to Definition 3.2, we can define a converge degree of Γ with respect to P by

$$c(P, \Gamma) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \sum_{j=1}^{s_i} p^2(X_i^j),$$

where s_i is the number of equivalence classes in $[\underline{P}X_i]$, $p(X_i^j) = \frac{|X_i^j|}{|X_i|}$ and X_i^j $(1 \le j \le s_i)$ is an equivalence class in $[\underline{P}X_i]$. It is clear that $0 \le c(P, \Gamma) \le 1$. In particular, $c(P, \Gamma)$ degenerates into $C(P, X)$ if $\Gamma = X$, i.e., $c(P, \Gamma) = \sum_{j=1}^{s} \frac{|X_i|^2}{|X|^2}$, where s is the number of equivalence classes in $[\underline{P}(X)]$, which represents the converge degree of X with respect to P.

In the following, we investigate two important properties of converge degree.

Theorem 3.3. *Let $S = (U, A)$ be an information system, $X \subseteq U$ and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \preceq R_2 \preceq \cdots \preceq R_n$ $(R_i \in 2^A)$. Let $P_i = \{R_1, R_2, \ldots, R_i\}$, then $\forall P_i$ $(i = 1, 2, \ldots, n)$, we have*

$$c(P_1, X) \le c(P_2, X) \le \cdots \le c(P_n, X).$$

Proof. From Theorem 3.2, it is clear that

$$[P_1(X)] \sqsubseteq [P_2(X)] \sqsubseteq \cdots \sqsubseteq [P_n(X)].$$

Suppose $1 \leq i < j \leq n$, $[\underline{P_i}(X)] = \{A_1, A_2, \ldots, A_m\}$ and $[\overline{P_j}(X)] = \{B_1, B_2, \ldots, B_n\}$, then $[P_i(X)] \sqsubseteq [P_j(X)]$ and $m > n$. That is to say, there may exist a partition $\{C_1, C_2, \ldots, C_n\}$ of $\{1, 2, \ldots, m\}$ such that $B_t = \bigcup_{l \in C_t} A_l$, $t = 1, 2, \ldots, n$. Therefore, one can obtain that

$$\begin{aligned} c(P_j, X) &= \frac{1}{|X|^2} \sum_{t=1}^{s_j} |B_t|^2 \\ &= \frac{1}{|X|^2} \sum_{t=1}^{s_j} \left| \bigcup_{l \in C_t} A_l \right|^2 = \frac{1}{|X|^2} \sum_{t=1}^{s_j} \left(\sum_{l \in C_t} |A_l| \right)^2 \\ &\geq \frac{1}{|X|^2} \sum_{t=1}^{s_j} \sum_{l \in C_t} |A_l|^2 = \frac{1}{|X|^2} \sum_{l=1}^{s_i} |A_l|^2 \\ &= c(P_i, X). \end{aligned}$$

Thus $c(P_1, X) \leq c(P_2, X) \leq \cdots \leq c(P_n, X)$. This completes the proof. □

Theorem 3.4. *Let $S = (U, A)$ be an information system, $\Gamma = \{X_1, X_2, \ldots, X_m\}$ a partition on U and $P = \{R_1, R_2, \ldots, R_n\}$ a family of attribute sets with $R_1 \preceq R_2 \preceq \cdots \preceq R_n$ ($R_i \in 2^A$). Let $P_i = \{R_1, R_2, \ldots, R_i\}$, then $\forall P_i$ ($i = 1, 2, \ldots, n$), we have*

$$c(P_1, \Gamma) \leq c(P_2, \Gamma) \leq \cdots \leq c(P_n, \Gamma).$$

Proof. It follows that from Theorem 3.3 that $c(P_1, X_i) \leq c(P_2, X_i) \leq \cdots \leq c(P_n, X_i)$ for any X_i $(i \leq m)$. Suppose $1 \leq k < t \leq n$, then $c(P_k, X_i) \leq c(P_t, X_i)$. Therefore, one can obtain that

$$\begin{aligned} c(P_k, \Gamma) &= \sum_{i=1}^{m} \frac{|X_i|}{|U|} \sum_{j=1}^{s_i} p^2(X_i^j) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \times c(P_k, X_i) \\ &\geq \sum_{i=1}^{m} \frac{|X_i|}{|U|} \times c(P_t, X_i) \\ &= c(P_t, \Gamma). \end{aligned}$$

Thus $c(P_1, \Gamma) \leq c(P_2, \Gamma) \leq \cdots \leq c(P_n, \Gamma)$. This completes the proof. □

Theorems 3.3 and 3.4 show that the converge degree of $X_i \in \Gamma$ with respect to P_i increases and the number of equivalence classes for describing the target concept X_i decreases as a granulation order becomes longer through adding equivalence relations. Since the converse approximation can clearly describe and simply the structures of the lower and upper approximation of a target concept on the basis of keeping the approximation measure, it may have some potential applications in rough set theory, such as description of multi-targets concepts, approximation classification and rule extraction from decision tables.

4. Rule extracting based on the converse approximation

Rough set theory is always used to mine some patterns in the form of "if ..., then ..." decision rules from decision tables. More exactly, the decision rules say that if some condition attributes have given values, then some decision attributes have other given values. In this section, as an application of the converse approximation, we apply this approach for decision-rule extracting from two types of decision tables (consistent decision tables and inconsistent decision tables). A rule-extracting algorithm based on the converse approximation called REBCA is designed to extract decision rules from a decision table, its time complexity is analysed and two illustrative examples are also employed to show the mechanism of algorithm REBCA.

Let $S = (U, C \cup D)$ with $C \cap D \neq \emptyset$ be an information system with decision attributes, we always call S a decision table, where C and D are condition and decision attribute sets respectively. If $C \preceq D$ ($U/\mathrm{IND}(C) \subseteq U/\mathrm{IND}(D)$), then S is a consistent decision table, otherwise we call it an inconsistent decision table [24,32]. In general, we design different rule-extracting algorithms according to these two kinds of decision tables. In the view of the converse approximation based on dynamic granulation, the decision classification induced by decision attributes can be regarded as the target classification (some target concepts) and the condition attribute sets can be used to construct a granulation order.

Let $S = (U, C \cup D)$ be a decision table, the significance of $c \in C - C'$ ($C' \subseteq C$) with respect to D is defined by

$$\mathrm{sig}_{C'}^{D}(c) = \gamma_{C' \cup \{c\}}(D) - \gamma_{C'}(D),$$

where $\gamma_{C'}(D) = \frac{1}{|U|}|pos_{C'}(D)| = \frac{1}{|U|}|\bigcup_{X \in U/\mathrm{IND}(D)} \underline{C'}X|$.

Generally, we can obtain some more useful decision rules by combining rules through deleting some condition attributes in decision tables. Based on the target classes, the converse approximation gives a mechanism that can be used to combine decision rules on the basis of keeping the confidence of each decision rule. One can know how the mechanism works from the following algorithm.

Algorithm. REBCA (for extracting decision rules from a decision table)

Input: decision table $S = (U, C \cup D)$;
Output: decision rules $Rule$.

(1) Compute decision classes $U/\mathrm{IND}(D) = \{X_1, X_2, \ldots, X_m\}$.

(2) From $j = 1$ to m Do
Let $C_1 \leftarrow C$, $P(1) \leftarrow \{\{C_1\}\}$.

(2.1) From $k = 1$ to $|C| - 1$ Do
For any $c \in C_k$, compute the significance $\mathrm{sig}_{C_k}^{D}(c)$.
Let $C_{k+1} \leftarrow C_k - c_0$, where c_0 : $\mathrm{sig}_{C_k}^{D}(c_0) = \min\{\mathrm{sig}_{C_k}^{D}(c),\ c \in C_k\}$.
Let $P(k+1) \leftarrow P(k) \cup \{\{C_{k+1}\}\}$;// $P(k+1)$ is the granulation order induced by C_k and C_{k+1};

(2.2) Compute $[\underline{P(|C|)}(X_j)]$;

(2.3) Put every decision rule $\mathrm{des}([x]) \rightarrow \mathrm{des}(X_j)$ into the rule base $\mathrm{rule}(j)$, where $[x] \in [\underline{P(|C|)}(X_j)]$.

(3) Output Rule $= \bigcup_{j=1}^{m} \mathrm{rule}(j)$.

Obviously, the generation of decision rules is not based on a reduct of a decision table but $X_j \in U/\mathrm{IND}(D)$ and a granulation order P_j induced by X_j and C in algorithm REBCA. Furthermore, the number of decision rules can be largely reduced on the basis of keeping that all decision rules are all certain rules in a decision table.

By using REBCA algorithm, the time complexity to extract decision rules from a decision table is polynomial.

At step (1), the time complexity for computing a decision partition is $O(|U|^2)$.

At step (2), we need to compute the complexity of each of three steps respectively.

For (2.1), Since $|C| - 1$ is the maximum value for the circle times, the time complexity for constructing P_j is

$$O((|C|-1)|U|^2 + (|C|-2)|U|^2 + \cdots + |U|^2) = O\left(\frac{|C|(|C|-1)}{2}|U|^2\right);$$

For (2.2), the time complexity for computing $[\underline{P(|C|)}(X_j)]$ is $O(|C||U|^2)$;

For (2.3), the time complexity for putting each decision rule into rule base is $O(|X_j|)$;

Therefore, the time complexity of step (2) is

$$\sum_{j=1}^{m}\left(O\left(\frac{|C|(|C|-1)}{2}|U|^2\right) + O(|C||U|^2) + O(|X_j|)\right) = O\left(\frac{m(|C|^2+|C|)}{2}|U|^2 + \sum_{j=1}^{m}|X_j|\right)$$
$$= O\left(\frac{m(|C|^2+|C|)}{2}|U|^2 + |U|\right).$$

At step (3), the time complexity is $O(|U|)$.

Table 1
A consistent decision table [18]

U	Attributes					
	a	b	c	d	e	f
1	3	2	3	0	2	1
2	2	2	3	0	2	1
3	1	0	2	0	1	1
4	3	1	3	0	2	1
5	2	0	3	0	2	1
6	0	0	1	0	0	0
7	3	2	0	1	1	0
8	1	0	1	0	0	0
9	2	0	2	1	1	0
10	1	1	3	1	0	0
11	1	0	2	0	1	1
12	3	2	0	1	1	0

Thus the time complexity of algorithm REBCA is

$$O(|U|^2) + O\left(\frac{m(|C|^2 + |C|)}{2}|U|^2 + |U|\right) + O(|U|) = O\left(\frac{m}{2}|C|^2|U|^2\right).$$

In fact, the time complexity of this algorithm can be reduced as $O(\frac{m}{2}|C|^2|U|log_2|U|)$ if we compute a classification by adopting ranking technique.

Remark. Algorithm REBCA is different from existing algorithms based on attribute reduct for extracting decision rules from decision tables. The time complexity of a rule-extracting algorithm based on attribute reduct is $O(|C|^3|U|^2)$. Obviously, the time complexity of algorithm REBCA is much smaller than those of the existing algorithms based on attribute reduct. In particular, its size partially depends on the number of decision classes m. In many practical applications, m is always smaller than the number of condition attributes $|C|$. Therefore, the time complexity of algorithm REBCA is largely reduced relative to those of the existing algorithms based on attribute reduct for extracting decision rules from decision tables.

In order to discuss applications of this algorithm, two types of decision tables will be examined, which are consistent decision tables and inconsistent decision tables, As follows, we show how algorithm REBCA to extract decision rules from decision tables by two illustrative examples. Firstly, we focus on a consistent decision table.

Example 4.1. A consistent decision table $S_1 = (U, C \cup D)$ is given by Table 1 [18], where $C = \{a, b, c, d\}$ is condition attribute set and $D = \{f\}$ is decision attribute set. By using algorithm REBCA, we can extract decision rules from Table 1.

By computing, it follows that

$$U/D = \{\{1, 2, 3, 4, 5, 11\}, \{6, 7, 8, 9, 10, 12\}\}.$$

Let $X_1 = \{1, 2, 3, 4, 5, 11\}$ and $X_2 = \{6, 7, 8, 9, 10, 12\}$. From REBCA, we have that $P = \{\{a, b, c, d, e\}, \{a, b, c, e\}, \{a, c, e\}, \{a, e\}, \{e\}\}$. From the definition of converse approximation, one can obtain that

$$[\underline{P}(X_1)] = \{\{3, 11\}, \{1, 2, 4, 5\}\}, \ [\underline{P}(X_2)] = \{\{7, 12\}, \{9\}, \{6, 8, 10\}\}.$$

Therefore, several decision rules can be extracted as follows:

$$\text{des}(\{1, 2, 4, 5\}) \rightarrow \text{des}(X_1), \quad \text{i.e., } (e = 2) \rightarrow (f = 1);$$
$$\text{des}(\{3, 11\}) \rightarrow \text{des}(X_1), \quad \text{i.e., } (a = 1,\ e = 2) \rightarrow (f = 1);$$
$$\text{des}(\{6, 8, 10\}) \rightarrow \text{des}(X_2), \quad \text{i.e., } (e = 0) \rightarrow (f = 0);$$
$$\text{des}(\{9\}) \rightarrow \text{des}(X_2), \quad \text{i.e., } (a = 2,\ e = 1) \rightarrow (f = 0);$$
$$\text{des}(\{7, 12\}) \rightarrow \text{des}(X_2), \quad \text{i.e., } (a = 3,\ e = 1) \rightarrow (f = 0).$$

For intuition, these five decision rules extracted by REBCA from the decision table S_1 are listed in Table 2.

Table 2
Decision rules extracted from the decision table S_1

Rule	Attributes		
	a	e	f
r_1		2	1
r_2	1	1	1
r_3		0	0
r_4	3	1	0
r_5	2	1	0

Table 3
An inconsistent decision table

U	Attributes				
	a_1	a_2	a_3	a_4	d
1	1	0	0	0	1
2	0	1	1	1	2
3	0	1	0	0	2
4	0	1	2	0	2
5	0	1	0	0	1
6	0	1	0	0	1

This example shows the mechanism of the decision-rule extracting algorithm based on the converse approximation in a consistent decision table. In fact, attribute set $\{a, e\}$ is a relative reduct of condition attribute set C in decision table S_1 (the notion of relative reduct is from the literature [32]) and these decision rules in Table 2 are also the same as the rule sets extracted from the algorithm based on attribute reduct. However, the time complexity of rule extracting algorithm based on the converse approximation is much smaller than the time complexity of the algorithm based on attribute reduct. □

Then, we investigate how to extract decision rules from an inconsistent decision table through using algorithm REBCA. The following example will be helpful in understanding its mechanism.

Example 4.2. An inconsistent decision table $S_2 = (U, C \cup D)$ is given by Table 3, where $C = \{a_1, a_2, a_3, a_4\}$ is condition attribute set and $D = \{d\}$ is decision attribute set. By using algorithm REBCA, we can extract decision rules from Table 3.

By computing, it follows that

$$U/D = \{\{1, 5, 6\}, \{2, 3, 4\}\}.$$

Let $X_1 = \{1, 5, 6\}$ and $X_2 = \{2, 3, 4\}$. From REBCA, we have that

$$P = \{\{a_1, a_2, a_3, a_4\}, \{a_2, a_3, a_4\}, \{a_2, a_3\}, \{a_3\}\}.$$

From the definition of converse approximation, one can obtain that

$$[\underline{P}(X_1)] = \{\{1\}\}, \qquad [\underline{P}(X_2)] = \{\{2, 4\}\}.$$

Therefore, two certain decision rules can be extracted as follows:

$$\mathrm{des}(\{1\}) \rightarrow \mathrm{des}(X_1), \quad \text{i.e., } (a_2 = 0,\ a_3 = 0) \rightarrow (d = 1);$$
$$\mathrm{des}(\{2, 4\}) \rightarrow \mathrm{des}(X_2), \quad \text{i.e., } (a_2 = 1,\ a_3 = 1) \rightarrow (d = 2).$$

And two uncertain decision rules are also acquired from the construction of $[\overline{P}(X_1)] - [\underline{P}(X_1)]$ and $[\overline{P}(X_2)] - [\underline{P}(X_2)]$ as follows:

$$\mathrm{des}(\{3, 5, 6\}) \rightarrow \mathrm{des}(X_1), \quad \text{i.e., } (a_2 = 1,\ a_3 = 0) \rightarrow (d = 2) \text{ with } \mu = \frac{1}{3};$$
$$\mathrm{des}(\{3, 5, 6\}) \rightarrow \mathrm{des}(X_2), \quad \text{i.e., } (a_2 = 1,\ a_3 = 0) \rightarrow (d = 1) \text{ with } \mu = \frac{2}{3};$$

Table 4
Decision rules extracted from the decision table S_2

Rule	Attributes		
	a_2	a_3	d
r_1	0	0	1
r_2	1	1	2
r_3	1	0	2
r_4	1	0	1

where μ denotes the uncertain measure of a decision rule [15]. For intuition, these four decision rules extracted by REBCA from the decision table S_2 are listed in Table 4.

This example shows the mechanism of the decision-rule extracting algorithm based on the converse approximation in inconsistent decision tables. In fact, attribute set $\{a_2, a_3\}$ is a relative reduct of condition attribute set C in this decision table (the notion of relative reduct is from the literature [32]), and these decision rules are also the same as rule sets obtained from the algorithm based on attribute reduct. Therefore, we can extract decision rules from an inconsistent decision table through using REBCA algorithm with smaller time complexity. □

From the above analyses, it is easy to see that by using algorithm REBCA one can extract decision rules from both a consistent decision table and an inconsistent decision table. Unlike some rule-extracting approaches based on attribute reduct, the time complexity of this algorithm is only $O(\frac{m}{2}|C|^2|U|\log_2|U|)$. Hence, this approach may be used to effectively extract decision rules from practical data sets.

5. Conclusions

In this paper we have extended rough set approximation under static granulation to rough set approximation under dynamic granulation, which is called the converse approximation of a target concept. The concept of converse approximation is mainly based on a granulation order from fine to coarse. For a subset of the universe, its converge degree is monotonously increasing as a granulation order becomes longer. This means that a target concept or a target classification can be approximated by the change in the converse approximation induced by a granulation order. As an application of the converse approximation, an algorithm called REBCA has been proposed for extracting decision rules from a decision table. Two illustrative examples (a consistent decision table and an inconsistent decision table) have been employed to show the validity of this algorithm. Moreover, the time complexity of algorithm REBCA is much smaller than those of existing rule-extracting approaches based on attribute reduct. These results will be helpful in efficiently extracting decision rules from practical large-scale data sets.

Acknowledgements

This work was partially supported by the High Technology Research and Development Program (No. 2007AA01Z165), the National Natural Science Foundation of China (No. 70471003, No. 60573074), the Foundation of Doctoral Program Research of the Ministry of Education of China (No. 20050108004), the Natural Science Foundation of Shanxi, China (No. 20031036), the Top Scholar Foundation of Shanxi, China, Key Project of Science and Technology Research of the Ministry of Education of China and the Graduate Student Innovation Foundation of Shanxi.

References

[1] Z. Pawlak, Rough sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publisher, London, 1991.
[2] G.Y. Wang, H. Yu, D.C. Dang, Decision table reduction based on conditional information entropy, Journal of Computers 25 (7) (2002) 1–8 (in Chinese).
[3] J. Bazan, J.F. Peters, A. Skowron, HS. Nguyen, M. Szczuka, Rough set approach to pattern extraction from classifiers, Electronic Notes in Theoretical Computer Science 82 (4) (2003) 1–10.
[4] Z. Pawlak, A. Skowron, Rudiments of rough sets, Information Sciences 177 (2007) 3–27.
[5] Z. Pawlak, A. Skowron, Rough sets: Some extensions, Information Sciences 177 (2007) 28–40.

[6] L.A. Zadeh, Fuzzy sets and Information Granularity, in: M. Gupta, R. Ragade, R. Yager (Eds.), Advances in Fuzzy Sets Theory and Application, North-Holland, Amsterdam, 1979, pp. 3–18.
[7] L.A. Zadeh, Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, Fuzzy Sets and Systems 90 (1997) 111–127.
[8] L.A. Zadeh, Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems, Soft Computing 2 (1) (1998) 23–25.
[9] Z. Pawlak, Granularity of knowledge, indiscernibility and rough sets, in: Proceedings of 1998 IEEE International Conference on Fuzzy Systems, 1998, pp. 106–110.
[10] Y.Y. Yao, Rough sets, neighborhood system and granular computing, in: Proceedings of 1999 IEEE Canadian Conference on Electrical and Computer Engineering, 1999, pp. 1553–1558.
[11] Y.Y. Yao, Stratified rough sets and granular computing, in: Proceedings of the 18th International Conference of the North American Fuzzy Information Processing Society, 1999, pp. 800–804.
[12] Y.Y. Yao, Granular computing: Basis issues and possible solutions, in: Proceedings of the Fifth International Conference on Computing and Information, vol. I, 2000, pp. 186–189.
[13] T.Y. Lin, Granular computing on binary relations I: Data mining and neighborhood systems, II: Rough set representations and belief functions, in: L. Polkowski, A. Skowron (Eds.), in: Rough Sets in Knowledge Discovery, vol. 1, 1998, pp. 107–140.
[14] Y.Y. Yao, Granular computing using neighborhood systems, in: R. Roy, T. Furuhashi, P.K. Chawdhry (Eds.), Advances in Soft Computing: Engineering Design and Manufacturing, 1999, pp. 539–553.
[15] G.J. Klir, Basic issues of computing with granular probabilities, in: Proceedings of 1998 IEEE International Conference on Fuzzy Systems, 1998, pp. 101–105.
[16] L. Zhang, B. Zhang, Theory of fuzzy quotient space (methods of fuzzy granular computing), Journal of Software 14 (4) (2003) 770–776 (in Chinese).
[17] J.Y. Liang, Z.Z. Shi, The information entropy, rough entropy and knowledge granulation in rough set theory, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 12 (1) (2004) 37–46.
[18] J.Y. Liang, Y.H. Qian, C.Y. Chu, D.Y. Li, J.H. Wang, Rough set approximation based on dynamic granulation, Lecture Notes in Computer Science 3641 (2005) 701–708.
[19] Y.H. Qian, J.Y. Liang, J. Wang, Rough set approximation under dynamic granulation, Computer Science 32 (3) (2005) 219–222 (in Chinese).
[20] W. Ziarko, Acquisition of hierarchy-structured probabilistic decision tables and rules from data, Expert Systems 20 (5) (2003) 305–310.
[21] W.X. Zhang, W.Z. Wu, J.Y. Liang, D.Y. Li, Theory and Method of Rough Sets, Science Press, Beijing, China, 2001.
[22] M. Beynon, Reducts within the variable precision rough sets model: A further investigation, European Journal of Operational Research 134 (3) (2001) 592–605.
[23] M. Kryszkiewicz, Comparative study of alternative type of knowledge reduction in inconsistent systems, International Journal of Intelligent Systems 16 (2001) 105–120.
[24] Y. Leung, D.Y. Li, Maximal consistent block technique for rule acquisition in incomplete information systems, Information Sciences 153 (2003) 85–106.
[25] J.Y. Liang, D.Y. Li, Uncertainty and Knowledge Acquisition in Information Systems, Science Press, Beijing, China, 2005.
[26] J.Y. Liang, Z.B. Xu, The algorithm on knowledge reduction in incomplete information systems, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 24 (1) (2002) 95–103.
[27] HS. Nguyen, D. Slezak, Approximation reducts and association rules correspondence and complexity results, Lecture Notes in Artificial Intelligence 1711 (1999) 137–145.
[28] M. Quafatou, α-RST: A generalization of rough set theory, Information Sciences 124 (2000) 301–316.
[29] A. Skwwron, C. Rauszer, The discernibility matrices and functions in information systems, in: R. Slowinski (Ed.), Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer Academic, Dordrecht, 1992, pp. 331–362.
[30] D. Slezak, Searching for dynamic reducts in inconsistent decision tables, in: Proceedings of IPMU'98, vol. 2, 1998, pp. 1362–1369.
[31] D. Slezak, Approximation reducts in decision tables, in: Proceedings of IPMU'96, vol. 3, 1996, pp. 1159–1164.
[32] W.X. Zhang, J.S. Mi, W.Z. Wu, Knowledge reductions in inconsistent information systems, International Journal of Intelligent Systems 18 (2003) 989–1000.
[33] W.Z. Wu, M. Zhang, H.Z. Li, J.S. Mi, Knowledge reduction in random information systems via dempster-shafer theory of evidence, Information Sciences 174 (2005) 143–164.