JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 24, 627-638 (2008)

Short Paper

Handling Redundancies for Disjunctive Information with Exclusive-or Semantics

NAN-CHEN HSIEH
Department of Information Management
National Taipei College of Nursing
Taipei, 112 Taiwan
E-mail: nchsieh@ntcn.edu.tw

Modeling disjunctive information and imprecise information are crucial problems in the field of database and artificial intelligence applications. This study aims in manipulating exclusive-or disjunctive information and imprecise information. The Contained Group among tuples is a promising construct to find out redundancies, and is a new method developed for dealing with redundancy problem.

Keywords: exclusive-or disjunctive information, imprecise information, contained group, redundancy

1. INTRODUCTION

The relational database model has been proven to be a very useful model in database management systems and has been widely applied [4]. The inherence of the relational database model is effective for precise and unambiguous data. However, real-world applications often involve incomplete information. In this study, a logical fuzzy relational database model was proposed to accommodate incomplete information into the relational model. Incomplete information means that instead of having a single value for an attribute, we have a subset of values for the corresponding attribute [9]. The first attempt to extend the relational model was null values. Null values are unknown or non-applicable values [3, 8, 9, 14]. Another attempt is to introduce disjunctive information into the relational database model [8, 10]. Following the closure assumptions in the presence of disjunctive information, some authors have been exploring the field of disjunctive logic programming [12].

Codd [3] used maybe tuples for manipulating existential null that exist in relational databases. Since we consider disjunctive information in this paper, maybe information [9] might be obtained from disjunctive information in two cases. One is when a database is update, and the other is when relational operators are applied to a database. Consider a relation containing the information about the age of employee and suppose that we want to represent the following disjunctive information in the relation, “John is either 28 or 30

Received March 9, 2006; revised May 24, 2006; accepted July 19, 2006.
Communicated by Ming-Syan Chen.

Table 1. The *EMPLOYEE* relation represented as a disjunctive database.

EMPLOYEE(*Name*, *Age*)
(John, 28) ∨ (John, 30)
(Ann, 28) ∨ (Peter, 28) ∨ (John, 28)

years old, " and "Anne, Peter or John are 28 years old." These pieces of disjunctive information can be represented as a disjunctive database in Table 1.

As pointed by Liu and Sunderraman [10], the treatment of disjunctive information is closely related to the treatment of maybe information. Suppose that at a later time the definite tuple "Anne is 28 years old" is added into the database. Since (Anne, 28) is true now, the definite tuple replaces the second disjunctive tuple. For the purpose of not losing information, this disjunctive tuple "(Peter, 28) ∨ (John, 28)" could be introduced as the form of maybe tuple into the database. However, since the disjunctive tuple "(John, 28) ∨ (John, 30)" already indicates that "(John, 28)" is possibly true. The newly introduced maybe tuple duplicates the information of the first disjunctive tuple. In this study, we intend to develop a new method for handling *exclusive-or disjunctive information* with redundancies.

For a better understanding of our presentation, this paper is organized as follows. Section 2 gives preliminaries of the logical fuzzy relational database model. Based on the introduction of contained group, section 3 proposes the solutions to redundancy problems regarding the logical fuzzy relation. Conclusions are finally made in section 4.

2. PRELIMINARIES

Reiter [14] proposed a generalized relational theory to specify the logical semantics of disjunctive information in the context of first-order logic. Liu and Sunderraman [10] proposed a generalization of relational database model on representing indefinite and maybe information. Vila *et al.* [15] and Yang [17] presented a logical definition of fuzzy relational databases. Hsieh [6, 7] and Wang *et al.* [16] proposed a logical definition of fuzzy relational databases to accommodate fuzzy indefinite and maybe information.

There are two kinds of disjunctive information: inclusive-or and exclusive-or. The disjunctive information is inclusive-or if more than one fact can be true simultaneously, while it is exclusive-or if only one fact can be true. In this paper, we consider exclusive-or disjunctive information in the logical fuzzy relational database model. Since we consider disjunctive information and imprecise information in this paper, the classical fuzzy relational database model must be modified to represent sure fuzzy disjunctive tuples and maybe fuzzy disjunctive tuples. In the following discussion, let *sure tuple* serves as an abbreviation for sure fuzzy disjunctive tuple, and let *maybe tuple* abbreviates maybe fuzzy disjunctive tuple.

Definition 1 Let $R(A_1, \ldots, A_n, \mu_r)$ be a logical fuzzy relational schema, where A_j is an attribute and $Dom(A_j)$ is the domain of attribute A_j, $j = 1, \ldots, n$. Then, a logical fuzzy relation contains two components r_{sure} and r_{maybe}, and $r = \langle r_{sure}, r_{maybe} \rangle$ over R, can be defined as follows,

$$r = \{(\boldsymbol{t}, \mu_r(\boldsymbol{t})) \mid (\boldsymbol{t}, \mu_r(\boldsymbol{t})) = \{(t_1, \mu_r(t_1)), \ldots, (t_k, \mu_r(t_k))\} \wedge k \geq 1 \wedge$$
$$t_i \in Dom(A_1) \times \ldots \times Dom(A_n) \wedge \mu_r(t_i) > 0 \wedge$$
$$\mu_r(t_i) = \min(\mu(t_i[A_1]), \ldots, \mu(t_i[A_n])) \wedge i = 1, \ldots, k\},$$

where $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ denotes a fuzzy disjunctive tuple, each $(t_i, \mu_r(t_i))$, $i = 1, \ldots, k$, is a sub-tuple or a possible interpretation of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, $\mu(t_i[A_j])$ denotes the membership value of tuple component $t_i[A_j]$ of attribute A_j with respect to sub-tuple $(t_i, \mu_r(t_i))$. ❑

In logical representation, the disjunctive form of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ is $(t_1, \mu_r(t_1)) \vee \ldots \vee (t_k, \mu_r(t_k))$, and $|\boldsymbol{t}|$ denotes the number of sub-tuples of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ or the length of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ in this paper. For each sure tuple $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, when $|\boldsymbol{t}| = 1$, that is, $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ contains only one sub-tuple, then $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ is a *definite sure tuple*; otherwise, $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ is an *indefinite sure tuple* in the logical fuzzy relation. Moreover, for each maybe tuple $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, when $|\boldsymbol{t}| = 1$, $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ is a *definite maybe tuple*, otherwise, $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ is an *indefinite maybe tuple* in the logical fuzzy relation. The semantics of fuzzy disjunctive tuple are described as follows.

Definition 2 Let $R(A_1, \ldots, A_n, \mu_r)$ be a logical fuzzy relational schema, DB be the logical fuzzy relational database and r be a logical fuzzy relation over R. Let $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ be a tuple in the logical fuzzy relation r, where $(\boldsymbol{t}, \mu_r(\boldsymbol{t})) = (t_1, \mu_r(t_1)) \vee \ldots \vee (t_k, \mu_r(t_k))$ and $k \geq 1$. The semantics of fuzzy exclusive-or disjunctive tuple describes as,

(1) When $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ is a sure tuple, that is, $DB \vdash r((\boldsymbol{t}, \mu_r(\boldsymbol{t})))$, then $DB \overset{p}{\vdash} r((t_j, \mu_r(t_j)))$, $j = 1, \ldots, k$, and there exists exactly one sub-tuple $(t_i, \mu_r(t_i))$, $(t_i, \mu_r(t_i)) \in (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, such that $DB \vdash r((t_i, \mu_r(t_i)))$.
(2) When $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ is a maybe tuple, that is, $DB \vdash r((\boldsymbol{t}, \mu_r(\boldsymbol{t})))$ or $DB \overset{p}{\vdash} r((\boldsymbol{t}, \mu_r(\boldsymbol{t})))$, then $DB \overset{p}{\vdash} r((t_j, \mu_r(t_j)))$, $(t_j, \mu_r(t_j)) \in (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, and there exists at most one sub-tuple $(t_i, \mu_r(t_i))$, $(t_i, \mu_r(t_i)) \in (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, such that $DB \vdash r((t_i, \mu_r(t_i)))$. ❑

In Definition 2, $\vdash$ stands for a tuple (or a sub-tuple) that can be derived from DB to a logical fuzzy relation and $\overset{p}{\vdash}$ stands for a tuple (or a sub-tuple) can be possibly derived from DB to a logical fuzzy relation, in logical interpretation.

Example 1: Consider the classical fuzzy relation *EMPLOYEE* in Table 2, where μ_r can be interpreted as a measure of possibility of the association between attribute values. Referring to the first tuple, the possibility that "John has *Age* = {28 or 30} is 0.67". For the rest of the tuples, the possibility that "Anne, Peter and John have *Age* = {28} are 0.86". In this fuzzy relation, all tuples are true but only with certain possibility. The classical fuzzy relational database model has a weakness that can not model clearly the concept of disjunction as described in Table 1.

Let *EMPLOYEE*(*Name*, *Age*, μ_r) be a logical fuzzy relational schema and r is a logical fuzzy relation over *EMPLOYEE*. As illustrated in Table 3, the logical fuzzy relation represents clearly the semantics of fuzzy disjunctive information. For example, a fuzzy fuzzy disjunctive tuple, "(Anne, 28, 0.86) ∨ (Peter, 28, 0.86) ∨ (John, 28, 0.86)," represents not only imprecision in attribute values but also that in the association between attribute values. This fuzzy disjunctive tuple presents that "Anne, Peter or John are 28 years old with possibility 0.86 and only one fact can be true." ❑

Table 2. The *EMPLOYEE* table represented as the classical type-2 fuzzy relation.

EMPLOYEE		
Name	*Age*	μ_r
John	28, 30	0.67
Anne	28	0.86
Peter	28	0.86
John	28	0.86

Table 3. The *EMPLOYEE* table represented as a logical fuzzy relation.

r (*Name*, *Age*, μ_r)
(John, 28, 0.67) ∨ (John, 30, 0.86)
(Anne, 28, 0.86) ∨ (Peter, 28, 0.86) ∨ (John, 28, 0.86)

3. HANDLING REDUNDANCIES

The key of preserving the beneficial properties of the classical fuzzy relational database model to the proposed logical fuzzy relational database model is to remove redundancies from the database. A tuple may include two kinds of redundancies-one occurs among sub-tuples of a tuple because they have the same attribute values, and the other occurs among tuples because some sub-tuples of a tuple can be derived from another tuple in the same logical fuzzy relation. This operation that eliminates redundancies will be repeated until the whole database is free of redundancies. In a logical fuzzy relation, we assume that two tuples have the same form, and then they indicate the same tuple. Therefore, for any two tuples ($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$) and ($\boldsymbol{t}_2$, $\mu_r(\boldsymbol{t}_2)$), when two tuples have the same form, the one with lower membership value is automatically eliminated from the logical fuzzy relation.

Definition 3 Let $R(A_1, \ldots, A_n, \mu_r)$ be a logical fuzzy relational schema, and r is a logical fuzzy relation over R. Let $(\boldsymbol{t}, \mu_r(\boldsymbol{t})) = (t_1, \mu_r(t_1)) \vee \ldots \vee (t_k, \mu_r(t_k))$ be a fuzzy disjunctive tuple, $(t_p, \mu_r(t_p))$ and $(t_q, \mu_r(t_q))$ be two sub-tuples of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, and α_i, ($i = 1, \ldots, n$) are the pre-defined threshold values associated with each attribute A_i. Then, two sub-tuples, $(t_p, \mu_r(t_p))$, $(t_q, \mu_r(t_q))$ are α-equal (denotes as $(t_p, \mu_r(t_p)) \approx (t_q, \mu_r(t_q))$) *if and only if*

$$E(t_p[A_i], t_q[A_i]) = true, \text{ and } \mu_{EQ}(\mu(t_p[A_i]), \mu(t_q[A_i])) \geq \alpha_i,\ i = 1, \ldots, n,$$

where E is an equality predicate, and the fuzzy resemblance relation EQ [13] is defined as:

$$\mu_{EQ}(x, y) = \begin{cases} 0 & x \neq y \\ 1 - abs(\mu(x) - \mu(y)) & x = y \end{cases}.$$

Since duplicate possible worlds disallow within a fuzzy disjunctive tuple. Then, for any fuzzy disjunctive tuple $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$,

$$((t_i, \mu_r(t_i)) \in (\boldsymbol{t}, \mu_r(\boldsymbol{t})) \wedge \neg\exists(t_j)((t_j, \mu_r(t_j)) \in (\boldsymbol{t}, \mu_r(\boldsymbol{t})) \wedge (t_i, \mu_r(t_i)) \approx (t_j, \mu_r(t_j))). \qquad \square$$

Consequently, let ($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$), ($\boldsymbol{t}_2$, $\mu_r(\boldsymbol{t}_2)$) be two fuzzy disjunctive tuples and let $\boldsymbol{t}_1 \subseteq \boldsymbol{t}_2$. Then, two fuzzy disjunctive tuples are α-equal (denotes as ($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$) ≈ ($\boldsymbol{t}_2$, $\mu_r(\boldsymbol{t}_2)$)) if and only if

$$((t_i, \mu_r(t_i)) \in (\boldsymbol{t}_1, \mu_r(t_1)) \wedge \exists(t_j)((t_j, \mu_r(t_j)) \in (\boldsymbol{t}_2, \mu_r(\boldsymbol{t}_2)) \wedge (t_i, \mu_r(t_i)) \approx (t_j, \mu_r(t_j))).$$

All the duplicate sub-tuples of lower membership degrees to other sub-tuples within a fuzzy disjunctive tuple should be removed. In Definition 3, the fuzzy resemblance relation is not a unique way of defining redundancies between tuples. For example, in a similarity-based or proximity-based fuzzy relation [1, 2], redundant tuples are collected into equivalence classes, and redundant tuples in the same equivalent class are merged to create a uniquely-determined fuzzy relation.

Example 2: Consider the logical fuzzy relation in Table 4. The fuzzy set descriptor "*young*", represents fuzzy values on attribute domain "*Age*", and α = 0.7 is the pre-defined threshold value associated with attribute "*Age*". Since (John, *young*, 0.67) ≈ (John, *young*, 0.86), the sub-tuples (John, *young*, 0.67) should be removed from the first tuple, and the first tuple became (John, *young*, 0.86). ❑

Table 4. The *EMPLOYEE* table represented as a logical fuzzy relation.

r (*Name*, *Age*, μ_r)
$\boldsymbol{t}_1$: (John, *young*, 0.67) ∨ (John, *young*, 0.86)
$\boldsymbol{t}_2$: (Anne, *young*, 0.86) ∨ (Peter, *young*, 0.86) ∨ (John, *young*, 0.86)

All tuples in a logical fuzzy relation may not assume to be without redundancy. Therefore, we define *minimal* of tuple in Definition 4 and the removal of tuples or sub-tuples are considered.

Definition 4 Let ($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$) be a fuzzy disjunctive tuple in the logical fuzzy relation. Then, ($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$) is said to be minimal *if and only if* there does not exist another fuzzy disjunctive tuple ($\boldsymbol{t}_2$, $\mu_r(\boldsymbol{t}_2)$) such that ($\boldsymbol{t}_2$, $\mu_r(\boldsymbol{t}_2)$) ≈ ($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$). ❑

Definition 5 Let ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) be a fuzzy disjunctive tuple in the logical fuzzy relation r, and ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) be a set of sub-tuples of ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$). Then,

(1) r ~ ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$), denotes that ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) is removed from r,
(2) ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) ~ ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$), denotes that a set of sub-tuples ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) of ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) is removed from (t, $\mu_r(t)$). ❑

For sure tuples, not only the membership values but also the degree of certainties must be considered. The certainty of sure tuples is measured by their length. More certain tuples are preferred in the resultant logical fuzzy relation. That is, in a logical fuzzy relation, if two sure tuples are α-equal, then it is implied that the sure tuple whose length is shorter, is the more certain tuple. Consequently, after redundancies among sure tuples are removed, the sure tuples whose length is closest to one are maintained. The notation for redundancies among sure tuples is describes as follows.

Definition 6 Let $R(A_1, \ldots, A_n, \mu_r)$ be a logical fuzzy relational schema, DB be the logical fuzzy relational database and r be a logical fuzzy relation over R. Let $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ be a tuple in the logical fuzzy relation r, $(\boldsymbol{t}', \mu_r(\boldsymbol{t}')) = \{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$ be a set of sub-tuples of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, and $r' = r_{sure} \sim (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$. Then,

(1) $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ is said to be a *strong redundant sure information* of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ *if and only if* a set of definite sure tuples, $(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))$, exists in r' such that $DB \vdash r'((t_i', \mu_r(t_i')))$, for $i = 1, \ldots, k$,

(2) $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ is said to be a *weak redundant sure information* of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ *if and only if* a set of definite sure tuples, $(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))$, exists in r' such that $DB \vdash^{p} r'((t_i', \mu_r(t_i')))$, $i = 1, \ldots, k$, and there exists a definite sure tuple, $(t, \mu_r(t))$, $(t, \mu_r(t)) \notin (\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ in r' such that $DB \nvdash^{p} r'(t, \mu_r(t))$, simultaneously. ❑

In the first case of Definition 6, when r contains a set of definite sure tuples, $\{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$, then $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ is duplicated and can be removed from r because more certain tuples are preferred. In the second case of Definition 6, we need $DB \vdash^{p} r'((t_i', \mu_r(t_i')))$, $i = 1, \ldots, k$, and $DB \nvdash^{p} r'(t, \mu_r(t))$ simultaneously, which implies all the derivable tuple of $\{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$, are sub-tuples of $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$. Suppose that we do not impose this constraint, it is possible that some tuples, which are not sub-tuples of $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$, are derivable from $\{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$. Furthermore, at least one sub-tuple of $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ can be derived from $\{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$, since $\{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$ are definite sure tuples.

Although we cannot make sure whether the derivable tuples from $\{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$, contain all sub-tuples of $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$. However, we do know that $DB \vdash r'((t_i', \mu_r(t_i')))$, $i = 1, \ldots, k$, simultaneously. In other words, all sub-tuples of $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ are possibly derived from $\{(t_1', \mu_r(t_1')), \ldots, (t_k', \mu_r(t_k'))\}$, therefore, information will not be lost. Accordingly, $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ is called the weak redundant sure information of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$. Definition 6 is the concept to find out redundancies among sure tuples. To extract available but hidden $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$, we define the *Sure Contained Group*, $SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$ of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ to the logical fuzzy relation r'.

Definition 7 Let $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ be a sure tuple in the logical fuzzy relation r, $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ be a set of sub-tuples of $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$, and $r' = r_{sure} \sim (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$. Then, $SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}'))) = \{(\boldsymbol{t}_1, \mu_r(\boldsymbol{t}_1)), \ldots, (\boldsymbol{t}_m, \mu_r(\boldsymbol{t}_m))\}$, is a *sure contained group* of $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ *if and only if*

(1) For each tuple $(\boldsymbol{t}_i, \mu_r(\boldsymbol{t}_i))$, $(\boldsymbol{t}_i, \mu_r(\boldsymbol{t}_i)) \in SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$, $\Rightarrow \boldsymbol{t}_i \subseteq \boldsymbol{t}' \wedge (\boldsymbol{t}_i, \mu_r(\boldsymbol{t}_i)) \in r_{sure}$,

(2) $((t_j', \mu_r(t_j')) \in (\boldsymbol{t}', \mu_r(\boldsymbol{t}')) \wedge (t_k', \mu_r(t_k')) \in (\boldsymbol{t}', \mu_r(\boldsymbol{t}')) \wedge t_j' \neq t_k' \wedge m \geq |\boldsymbol{t}'| \wedge (t_j', \mu_r(t_j'))$ and $(t_k', \mu_r(t_k'))$ can not belong to only one tuple, $(\boldsymbol{t}_i, \mu_r(\boldsymbol{t}_i))$, $(\boldsymbol{t}_i, \mu_r(\boldsymbol{t}_i)) \in SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}'))))$,

(3) $\boldsymbol{t}' = \{t_i \mid (\exists \boldsymbol{t}_j)\ ((\boldsymbol{t}_j, \mu_r(\boldsymbol{t}_j)) \in SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}'))) \wedge t_i \in t_j \wedge |t_i| = 1)\}$. ❑

Since the tuple whose length is more close to one is preferred, each tuple $(\boldsymbol{t}_i, \mu_r(\boldsymbol{t}_i))$, $i = 1, \ldots, m$, in $SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$ and $\boldsymbol{t}_i \subseteq \boldsymbol{t}'$ is required. It is possible that more than one sure contained groups in (**t**', μ_r(**t**')) exist in the logical fuzzy relation, *r*'. According to Definition 8, it is not required to consider the tuples that are not in $SCG_{r'}$((**t**', μ_r(**t**'))) to find redundant information. Consequently, there does not exist any tuples, which are not sub-tuples of (**t**', μ_r(**t**')), that can be derived from $SCG_{r'}$((**t**', μ_r(**t**'))). The membership values of sub-tuples in the resultant sure tuples are chosen as minimum of all the sub-tuples that are α-equal in the contained group. Definition 7 presents that all sub-tuples of (**t**', μ_r(**t**')) could be derived from the sure tuples in $SCG_{r'}$((**t**', μ_r(**t**'))), simultaneously.

Example 3: Let *EMPLOYEE*(*Name*, *Age*, μ_r) be a logical fuzzy relational schema and *r* is a logical fuzzy relation over *EMPLOYEE*:

Table 5. A logical fuzzy relation with redundancies.

r (*Name*, *Age*, μ_r)
$\boldsymbol{t}_1$: (Anne, *young*, 1.0) ∨ (John, *young*, 0.86)
$\boldsymbol{t}_2$: (John, *young*, 0.86) ∨ (Mary, *young*, 1.0)
$\boldsymbol{t}_3$: (Anne, *young*, 0.86) ∨ (Peter, *young*, 0.86) ∨ (John, *young*, 0.86)

Table 6. A logical fuzzy relation without redundancy.

r (*Name*,*Age*,μ_r)
t_1: (Anne, *young*, 1.0) ∨ (John, *young*, 0.86)
t_2: (John, *young*, 0.86) ∨ (Mary, *young*, 1.0)
t_3' (Peter, *young*, 0.86)

Consider the redundant information of ($\boldsymbol{t}_3$, $\mu_r(\boldsymbol{t}_3)$), let (**t**', μ_r(**t**')) ={(Anne, *young*, 0.86), (John, *young*, 0.86)} be a set of sub-tuples of ($\boldsymbol{t}_3$, $\mu_r(\boldsymbol{t}_3)$) and *r*' = *r* ~ ($\boldsymbol{t}_3$, $\mu_r(\boldsymbol{t}_3)$). $SCG_{r'}$((**t**', μ_r(**t**'))) = {($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$)} is a sure contained group of (**t**', μ_r(**t**')) to *r*'. Therefore, (t_3, $\mu_r(t_3)$) is the redundant information, and ($\boldsymbol{t}_3$, $\mu_r(\boldsymbol{t}_3)$) is removed from *r* and further reduced to a maybe tuple, (Peter, *young*, 0.86). Accordingly, the remaining sure tuples can not find any sure contained group, and the membership values of sub-tuples in the resultant sure tuples are updated as shown in Table 6. ❑

Next, we present the redundancies among maybe tuples. Similar to the definition of sure contained group, the *maybe contained group*, $MCG_{r'}$((**t**', μ_r(**t**'))), of the tuple, (**t**', μ_r(**t**')), to *r*', is defined as follows.

Definition 8 Let (**t**, μ_r(**t**)) be a maybe tuple in the logical fuzzy relation *r*, (**t**', μ_r(**t**')) be a set of sub-tuples of (**t**, μ_r(**t**)), and *r*' = *r* ~ (**t**, μ_r(**t**)). Then, (**t**', μ_r(**t**')) is said to be the redundant information of (*t*, $\mu_r(t)$), *if and only if* there exists a fuzzy disjunctive tuple, (**t**'', μ_r(**t**'')) and **t**' ⊆ **t**'', in *r*' such that all sub-tuples of (**t**'', μ_r(**t**'')), can be derived from *r*' simultaneously. ❑

Definition 9 Let (**t**, μ_r(**t**)) be a maybe tuple in the logical fuzzy relation *r*, (**t**', μ_r(**t**')) be a set of sub-tuples of (**t**, μ_r(**t**)), and *r*' = *r* ~ (**t**, μ_r(**t**)). Then $MCG_{r'}$((**t**', μ_r(**t**'))) = {($\boldsymbol{t}_1$, $\mu_r(\boldsymbol{t}_1)$), …, ($\boldsymbol{t}_m$, $\mu_r(\boldsymbol{t}_m)$)}, is a *maybe contained group* of (**t**', μ_r(**t**')) *if and only if*

(1) For each tuple ($\boldsymbol{t}_i$, $\mu_r(\boldsymbol{t}_i)$), ($\boldsymbol{t}_i$, $\mu_r(\boldsymbol{t}_i)$) $\in MCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$,
$\Rightarrow (\boldsymbol{t}_i \cap \boldsymbol{t}' \neq \emptyset) \vee (\boldsymbol{t}_i \cap \boldsymbol{t}_j \neq \emptyset) \wedge (\boldsymbol{t}_i, \mu_r(\boldsymbol{t}_i)) \in r$, $1 \leq i \leq m$, $1 \leq j \leq m$, $\wedge\ i \neq j$,
(2) $\boldsymbol{t}' \subseteq \{t_i \mid (\exists \boldsymbol{t}_j)((\boldsymbol{t}_j, \mu_r(\boldsymbol{t}_j)) \in MCG_r((\boldsymbol{t}', \mu_r(\boldsymbol{t}'))) \wedge t_i \in \boldsymbol{t}_j \wedge |t_i| = 1)\}$,
(3) $|\boldsymbol{t}'| > 1 \Rightarrow ((t'_j, \mu_r(t'_j)) \in (\boldsymbol{t}', \mu_r(\boldsymbol{t}')) \wedge (t'_k, \mu_r(t'_k)) \in (\boldsymbol{t}', \mu_r(\boldsymbol{t}')) \wedge t'_j \neq t'_k \wedge (t'_j, \mu_r(t'_j))$ and $(t'_k, \mu_r(t'_k))$ can not belong to only one tuple ($\boldsymbol{t}_i$, $\mu_r(\boldsymbol{t}_i)$), ($\boldsymbol{t}_i$, $\mu_r(\boldsymbol{t}_i)$) $\in MCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}'))))$ $\vee$ $|\boldsymbol{t}'| = 1 \Rightarrow ((t'_j, \mu_r(t'_j)) \in (\boldsymbol{t}', \mu_r(\boldsymbol{t}')) \wedge (t'_j, \mu_r(t'_j))$ can not belong to only one tuple, ($\boldsymbol{t}_i$, $\mu_r(\boldsymbol{t}_i)$), ($\boldsymbol{t}_i$, $\mu_r(\boldsymbol{t}_i)$) $\in r'_{sure}$). ❑

Definition 9 presents that all sub-tuples of ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) can be derived from the maybe tuples in $MCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$, simultaneously.

The maybe contained group is used to find out redundancies among maybe tuples. Similar to the sure contained group, if ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) is determined as redundant information to the maybe tuple ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$), then ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) is removed from the logical fuzzy relation and ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) ~ ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) is added as a new maybe tuple. In order to reserve maybe tuples with better quality, let (t_i, $\mu_r(t_i)$), (t_j, $\mu_r(t_j)$) be two sub-tuples of ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$), ($\boldsymbol{t}_k$, $\mu_r(\boldsymbol{t}_k)$) respectively, and (t_i, $\mu_r(t_i)$) $\approx$ (t_j, $\mu_r(t_j)$), then the membership value of each sub-tuple (t_i, $\mu_r(t_i)$) of ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$), should be compared to that sub-tuple (t_j, $\mu_r(t_j)$) of ($\boldsymbol{t}_k$, $\mu_r(\boldsymbol{t}_k)$), where ($\boldsymbol{t}_k$, $\mu_r(\boldsymbol{t}_k)$) $\in SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$. Finally, the maximum membership value of these two sub-tuples is reserved.

The *REDUCE* operation is used to eliminate the redundancies over a logical fuzzy relation. Instead of the performance, we only consider the semantics of the fuzzy disjunctive information.

Algorithm 1 *REDUCE* operation
Input: A logical fuzzy relation r.
Output: $r = REDUCE(r)$.
Method:

Reorder the tuples in the logical fuzzy relation, r, by length, and
  let ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) be the tuple in the logical fuzzy relation r;
Repeat
  For each sure tuple, ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$), in r Do
  {Finding a possible tuple ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$), and ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) is a set of sub-tuples of ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$);
   Let $r' = r_{sure} \sim (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$;
   If the sure contained group, $SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$, exists Then
    {For all (t_i, $\mu_r(t_i)$), (t_i, $\mu_r(t_i)$) $\in$ ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) Do
     {Let (t_j, $\mu_r(t_j)$) $\in$ ($\boldsymbol{t}_k$, $\mu_r(\boldsymbol{t}_k)$), ($\boldsymbol{t}_k$, $\mu_r(\boldsymbol{t}_k)$) $\in SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$ and
       (t_i, $\mu_r(t_i)$) $\approx$ (t_j, $\mu_r(t_j)$);
      $\mu_r(t_i) = \min(\mu_r(t_i), \mu_r(t_j))$; }}
    add ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$)~($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) into r as a maybe tuple;
    delete ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) from r;}
Until all sure tuples in r are minimal;
Repeat
  For each maybe tuple, ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$) in r Do
  {Finding a possible tuple ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$), and ($\boldsymbol{t}'$, $\mu_r(\boldsymbol{t}')$) is a set of sub-tuples of ($\boldsymbol{t}$, $\mu_r(\boldsymbol{t})$);
   Let $r' = r \sim (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$;

If the maybe contained group, $MCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$, exists Then
{For all $(t_i, \mu_r(t_i))$, $(t_i, \mu_r(t_i)) \in (\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ Do
{Let $(t_j, \mu_r(t_j)) \in (\boldsymbol{t}_k, \mu_r(\boldsymbol{t}_k))$, $(\boldsymbol{t}_k, \mu_r(\boldsymbol{t}_k)) \in MCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}')))$ and
$(t_i, \mu_r(t_i)) \approx (t_j, \mu_r(t_j))$;
$\mu_r(t_i) = \max(\mu_r(t_i), \mu_r(t_j))$; }}
add $(\boldsymbol{t}, \mu_r(\boldsymbol{t})) \sim (\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ into r as a maybe tuple;
delete $(\boldsymbol{t}, \mu_r(\boldsymbol{t}))$ from r; }
Until all maybe tuples in r are minimal;
End *REDUCE* Operation.

Cubero *et al.* [5] and Medina *et al.* [11] suggested that it is not necessary to build an ad hoc database system storing fuzzy disjunctive information. A general relational database management system can be used to establish special relations and dictionaries to manage fuzzy information. Efficient data management can thus be exploited. Let n_s, n_m be the number of tuples in r_{sure} and r_{maybe}, respectively. Then, the time complexity of the *REDUCE* operation is $O((n_s^2 + n_m * (n_s + n_m))$. In this paper, instead of considering the performance of this algorithm, we concerned about the semantics of the *REDUCE* operation. The performance of this algorithm can be improved by optimization mechanism of the corresponding *DBMS*.

Example 4: Let *EMPLOYEE*(*Name*, *Age*, μ_r) be a logical fuzzy relation schema and r is a logical fuzzy relation over *EMPLOYEE*:

Table 7. A logical fuzzy relation with sure and maybe tuples.

r (*Name*, *Age*, μ_r)
$\boldsymbol{t}_1$: (Anne, *middle*, 0.8) ∨ (Peter, *young*, 1) ∨ (John, *young*, 0.8) ∨ (Mary, *middle*, 1)
$\boldsymbol{t}_2$: (Anne, *middle*, 0.9) ∨ (Peter, *young*, 0.9)
$\boldsymbol{t}_3$: (Peter, *young*,1) ∨ (John, *young*, 0.8)
$\boldsymbol{t}_4$: (Anne, *middle*, 0.8)
$\boldsymbol{t}_5$: (Anne, *middle*, 0.8) ∨ (Mary, *middle*, 0.9) ∨ (Ada, *young*, 1)
$\boldsymbol{t}_6$: (Mary, *middle*, 0.8) ∨ (Ada, *young*, 1)
$\boldsymbol{t}_7$: (Mary, *middle*, 0.9) ∨ (Alex, *middle*, 1)
$\boldsymbol{t}_8$: (Ada, *young*, 1) ∨ (Alex, *middle*, 0.9)

In this example, we assume all the thresholds values associated with each attribute are 0.8. We present how to eliminate redundant information among sure tuples. Consider the redundant information of $(\boldsymbol{t}_1, \mu_r(\boldsymbol{t}_1))$, let $r' = r_{sure} \sim (\boldsymbol{t}_1, \mu_r(\boldsymbol{t}_1))$ and choose $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ = (Anne, *middle*, 0.8) ∨ (Peter, *young*, 1) ∨ (John, *young*, 0.8). Since there exists a sure contained group, $SCG_{r'}((\boldsymbol{t}', \mu_r(\boldsymbol{t}'))) = \{(\boldsymbol{t}_2, \mu_r(\boldsymbol{t}_2)), (\boldsymbol{t}_3, \mu_r(\boldsymbol{t}_3)), (\boldsymbol{t}_4, \mu_r(\boldsymbol{t}_4))\}$, of $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ to r'. Therefore, $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ is the redundant information to $(\boldsymbol{t}_1, \mu_r(\boldsymbol{t}_1))$ and $(\boldsymbol{t}_1, \mu_r(\boldsymbol{t}_1))$ is removed from r. Then, the tuple, $(\boldsymbol{t}_1, \mu_r(\boldsymbol{t}_1)) \sim (\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ = (Mary, *middle*, 1), should be added to r as a maybe tuple. Similarly, $(\boldsymbol{t}_2, \mu_r(\boldsymbol{t}_2))$ is removed from r and (Peter, *young*, 0.9) should be added to r as a maybe tuple.

Next, we present how to eliminate redundant information among maybe tuples. Consider the redundant information of $(\boldsymbol{t}_5, \mu_r(\boldsymbol{t}_5))$ and choose $(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))$ = {(Anne, *middle*, 0.8), (Mary, *middle*, 0.9), (Ada, *young*, 1)}. Let $r' = r \sim (\boldsymbol{t}_5, \mu_r(\boldsymbol{t}_5))$. Since the maybe contained group $MCG_{r'}((\boldsymbol{t}_5, \mu_r(\boldsymbol{t}_5))) = \{(\boldsymbol{t}_1', \mu_r(\boldsymbol{t}_1')), (\boldsymbol{t}_4, \mu_r(\boldsymbol{t}_4)), (\boldsymbol{t}_6, \mu_r(\boldsymbol{t}_6)), (\boldsymbol{t}_7, \mu_r(\boldsymbol{t}_7)), (\boldsymbol{t}_8, \mu_r(\boldsymbol{t}_8))\}$ exists, maybe tuple $(\boldsymbol{t}_5, \mu_r(\boldsymbol{t}_5))$ is removed from r. Similarly, since $MCG_{r'}((\boldsymbol{t}_6, \mu_r(\boldsymbol{t}_6))) = \{(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))\}$ exists, $(\boldsymbol{t}_6, \mu_r(\boldsymbol{t}_6))$ is also removed from r, and (Ada, *young*, 1.0) is added as a maybe tuple. $MCG_{r'}((\boldsymbol{t}_7, \mu_r(\boldsymbol{t}_7))) = \{(\boldsymbol{t}', \mu_r(\boldsymbol{t}'))\}$ exists. $(\boldsymbol{t}_7, \mu_r(\boldsymbol{t}_7))$ is removed and (Alex, *middle*, 1) is added as a maybe tuple. Similarly, $(\boldsymbol{t}_8, \mu_r(\boldsymbol{t}_8))$ can be reduced as a maybe tuple (Ada, *young*, 1). The remaining maybe tuples do not exist in the maybe contained group anymore. Table 9 is a logical fuzzy relation without redundancy. Tuple $(\boldsymbol{t}_2', \mu_r(\boldsymbol{t}_2'))$ can not be removed, because after0 removed, sub-tuples (Peter, *young*, 1) and (John, *young*, 0.8) in $(\boldsymbol{t}_3, \mu_r(\boldsymbol{t}_3))$ could not be derived from the logical fuzzy relational database simultaneously, which make a contradiction to Definition 2. ❑

Table 8. Elimination the redundancies among sure tuples.

r (*Name*, *Age*, μ_r)
$\boldsymbol{t}_3$: (Peter, *young*, 1) ∨ (John, *young*, 0.8)
$\boldsymbol{t}_4$: (Anne, *middle*, 0.8)
$\boldsymbol{t}_5$: (Anne, *middle*, 0.8) ∨ (Mary, *middle*, 0.9) ∨ (Ada, *young*, 1)
$\boldsymbol{t}_6$: (Mary, *middle*,0.8) ∨ (Ada, *young*, 1)
$\boldsymbol{t}_7$: (Mary, *middle*,0.9) ∨ (Alex, *middle*, 1)
$\boldsymbol{t}_8$: (Ada, *young*, 1) ∨ (Alex, *middle*, 0.9)
$\boldsymbol{t}_1$': (Mary, *middle*, 1)
$\boldsymbol{t}_2$': (Peter, *young*, 0.9)

Table 9. Elimination the redundancies among maybe tuples.

r (*Name*, *Age*, μ_r)
$\boldsymbol{t}_3$: (Peter, *young*, 1) ∨ (John, *young*, 0.8)
$\boldsymbol{t}_4$: (Anne, *middle*, 0.8)
$\boldsymbol{t}_1$': (Mary, *middle*, 1)
$\boldsymbol{t}_2$': (Peter, *young*, 0.9)
$\boldsymbol{t}_7$': (Alex, *middle*, 1)
$\boldsymbol{t}_8$': (Ada, *young*, 1)

4. CONCLUSION

We present the concept of the Contained Group among tuples to find redundancies. This is a proposed method of updating databases to eliminate redundancies among fuzzy exclusive-or disjunctive tuples. After reorganizing the logical fuzzy relation, the logical fuzzy relation will be uniquely-determined. A complete set of relational algebra and the practical applications of the logical fuzzy relational model will be proposed in the future.

5. REFERENCES

1. B. P. Buckles and F. E. Petry, "Extending the fuzzy database with fuzzy number,"

Information Science, Vol. 34, 1984, pp. 145-155.
2. B. P. Buckles and F. E. Petry, "Information-theoretical characterization of fuzzy relational databases," *IEEE Transactions on System*, *Man*, *Cybernetics*, Vol. 13, 1983, pp. 72-77.
3. E. F. Codd, "Extending the database relational model to capture more meaning," *ACM Transactions on Database Systems*, Vol. 4, 1979, pp. 397-434.
4. E. F. Codd, "A relational model for large shared data banks," *Communications of the ACM*, Vol. 13, 1970, pp. 377-387.
5. J. C. Cubero, J. M. Medina, O. Pons, and M. A. Vila, "Data summarization in relational databases through fuzzy dependencies," *Information Sciences*, Vol. 121, 1999, pp. 233-270.
6. N. C. Hsieh, "Answers to database queries concerning imprecise and uncertain information in logical fuzzy relational databases," *International Journal of Intelligent Systems*, Vol. 20, 2005, pp. 647-668.
7. N. C. Hsieh, "Handling indefinite and maybe information in logical fuzzy relational databases," *International Journal of Intelligent Systems*, Vol. 19, 2004, pp. 257-276.
8. T. Imielinski and K. Vadaparty, "Complexity of query processing in databases with or-objects," in *Proceedings of the 8th ACM SIGACT-SIGMOD Symposium on Principles of Database Systems*, 1989, pp. 51-65.
9. W. Lipski, "On semantic issues connected with incomplete information systems," *ACM Transactions on Database Systems*, Vol. 4, 1979, pp. 262-296.
10. K. C. Liu and R. Sunderraman, "A generalized relational model for indefinite and maybe information," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 3, 1991, pp. 65-77.
11. J. M. Medina, O. Pons, and M. A. Vila, "GEFRED: a generalized model for fuzzy relational databases," *Information Sciences*, Vol. 76, 1994, pp. 87-109.
12. J. Minker, "Toward a foundation of disjunctive logic programming," in *Proceedings of the North American Logic Programming Conference*, 1989, pp. 1215-1235.
13. K. V. S. V. N. Raju and A. K. Majumdar, "Fuzzy functional dependencies and lossless join decomposition of fuzzy relational database systems," *ACM Transactions on Database Systems*, Vol. 13, 1988, pp. 129-166.
14. R. Reiter, "A sound and sometimes complete query evaluation algorithm for relational databases with null values," *Journal of the ACM*, Vol. 33, 1986, pp. 349-370.
15. M. A. Vila, J. C. Cubero, J. M. Medina, and O. Pons, "Uncertain fuzzy values still in the framework of first order logic," *International Journal of Intelligent Systems*, Vol. 17, 2002, pp. 873-886.
16. M. H. Wang, H. C. Keh, N. C. Hsieh, and T. S. Wang, "A study of disjunctive information in fuzzy relational databases," *Journal of Information Science and Engineering*, Vol. 22, 2006, pp. 199-213.
17. J. D. Yang, "Implicit predicates for handling disjunctive fuzzy information in fuzzy databases," *International Journal of Intelligent Systems*, Vol. 17, 2002, pp. 1085-1100.

Nan-Chen Hsieh (謝楠楨) received the Ph.D. degree in Computer Science and Information Engineering from the Tamkang University, Taiwan, in 1996. He is currently a Professor in the Department of Information Management at National Taipei College of Nursing, Taiwan. His research interests are in the area of data mining