Handling Inconsistency of Vague Relations with Functional Dependencies

An Lu and Wilfred Ng

Department of Computer Science and Engineering
The Hong Kong University of Science and Technology
Hong Kong, China
{anlu, wilfred}@cse.ust.hk

Abstract. Vague information is common in many database applications due to internet-scale data dissemination, such as those data arising from sensor networks and mobile communications. We have formalized the notion of a vague relation in order to model vague data in our previous work. In this paper, we utilize Functional Dependencies (FDs), which are the most fundamental integrity constraints that arise in practice in relational databases, to maintain the consistency of a vague relation. The problem we tackle is, given a vague relation r over a schema R and a set of FDs F over R, what is the "best" approximation of r with respect to F when taking into account of the median membership (m) and the imprecision membership (i) thresholds. Using these two thresholds of a vague set, we define the notion of mi-overlap between vague sets and a merge operation on r. Satisfaction of an FD in r is defined in terms of values being mi-overlapping. We show that Lien's and Atzeni's axiom system is sound and complete for FDs being satisfied in vague relations. We study the chase procedure for a vague relation r over R, named VChase(r, F), as a means to maintain consistency of r with respect to F. Our main result is that the output of the procedure is the most object-precise approximation of r with respect to F. The complexity of VChase(r, F) is polynomial time in the sizes of r and F.

1 Introduction

Fuzzy set theory has long been introduced to handle inexact and imprecise data by Zadeh's seminal paper in [1]. In fuzzy set theory, each object $u \in U$ is assigned a single real value, called the *grade of membership*, between zero and one. (Here U is a classical set of objects, called the *universe of discourse*.) In [2], Gau et al. point out that the drawback of using the single membership value in fuzzy set theory is that the evidence for $u \in U$ and the evidence against $u \in U$ are in fact mixed together. In order to tackle this problem, Gau et al. propose the notion of *Vague Sets* (VSs), which allow using interval-based membership instead of using point-based membership as in FSs. We have shown in our previous work [3] that the interval-based membership generalization in VSs is more expressive in capturing vague data semantics.

In a vague relation, each object with a *vague membership* belongs to a VS. A vague membership (also called a vague value) is a subinterval $[\alpha(u), 1-\beta(u)]$ of the unit interval [0,1], where $0 \leq \alpha(u) \leq 1-\beta(u) \leq 1$. A true (false) membership function $\alpha(u)$

$(\beta(u))$ is a lower bound on the grade of membership of u derived from the evidence for (against) u.

In order to compare two vague values, we define the *median membership*, $M_m = (\alpha + 1 - \beta)/2$, which represents the overall evidence contained in a vague value, and the *imprecision membership*, $M_i = (1 - \beta - \alpha)$, which represents the overall imprecision of a vague value. With M_m and M_i, we have the one-to-one correspondence between a vague value, denoted by $[\alpha, 1 - \beta]$, and a mi*-pair* vague value, denoted by $< M_m, M_i >$, for a given object. We further extend the notion of mi-overlap to VSs.

Integrity constraints ensure that changes made to the database do not result in a loss of data consistency. The notion of a Functional Dependency (FD) [4], the most fundamental integrity constraints, being satisfied in a vague relation can be formalized in terms values being mi-overlapping rather than equal. We show that Lien's and Atzeni's axiom system [5, 4] is sound and complete for FDs being satisfied in vague relations. A vague relation is said to be consistent with respect to a set of FDs F if it satisfies F. We define the chase procedure for a vague relation r over R, named $VChase(r, F)$, to tackle the consistency problem with respect to F, defined on vague relations [3]. Our main result is that the output of the procedure is the most $object$-precise (or O-precise in our notation) approximation of r with respect to F.

Here we give a motivating example. Consider a vague relation schema $R = \{S, T\}$, where S stands for the evidence of a sensor ID and T stands for the temperature monitored by a sensor. Here S and T are vague concepts, their values are all represented by VSs. Suppose the attributes S and T share the common universes of discourse, $U = \{0, 1, \ldots, 10\}$. A vague relation r_1 over R is shown in Table 1, where the attributes S and T are vague. The VS <0.8,0.1>/0 means the evidence for "the sensor ID is 0" is 0.8 and the imprecision for it is 0.1. The median membership threshold C and the imprecision membership threshold I are called the mi*-thresholds*. For simplicity, we only show the elements in the values of S and T that satisfy the mi-thresholds. Intuitively, this means that the elements in the relation all have strong evidence relative to the thresholds. The saying that two VSs mi-overlap means they have at least one common object which satisfies the mi-thresholds (i.e., $0.8 \geq C$ and $0.1 \leq I$ in this example). We regard two mi-overlapping VSs are similar to each other to some extent and extend the classical FD concept to vague relations. Suppose that the FD $S \rightarrow T$ is specified as a constraint, meaning that same sensor reads same temperature in a vague sense.

We assume a vague relation r_1 over R, where the current temperature may be obtained from different sensors. Thus, at any given time the information may be inconsistent. It can be verified that r_1 satisfies $S \rightarrow T$ and is consistent. Suppose later a vague tuple was inserted into r_1, we have the vague relation r_2 shown in Table 2. It can be verified that r_2 does not satisfy $S \rightarrow T$ and is inconsistent, since the evidence of S shows that the two tuples have the common object 0 mi-overlapped, but the values of T do not have a common object and thus do not mi-overlap. The vague relation r_2 can be approximated by the less O-precise relation r_3, shown in Table 3. It can be verified that r_3 satisfies $S \rightarrow T$ and is consistent. The vague relation r_3 (one tuple) is in fact the most O-precise approximation of r_2. The transformation from r_2 to r_3 is based on the VChase procedure introduced later.

Table 1. Sensor relation r_1

S	**T**
<0.8,0.1>/0	<0.9,0>/0

Table 2. Sensor relation r_2

S	**T**
<0.8,0.1>/0	<0.9,0>/0
<0.9,0.2>/0	<0.8,0.1>/1

Table 3. Sensor relation r_3

S	**T**
<0.9,0.1>/0	<0.9,0>/0 + <0.8,0.1>/1

We define the merge operation which replaces each attribute value in r by the mi-union of all attribute values with respect to the same reflexive and transitive closure under mi-overlap. This leads to a partial order on merged vague relations and the notion of a vague relation being less O-precise than another vague relation. This partial order induces a lattice on the set of merged vague relations, which we denote by $MERGE(R)$, based on $object$-equivalence (O-equivalence for short) classes. We define the VChase procedure for a vague relation r over R as a means of maintaining consistency of r with respect to F. We investigate the properties of the VChase procedure showing amongst other results that it outputs a consistent vague relation. The output of VChase is unique. VChase can be computed in polynomial time in the sizes of r and F, and the procedure commutes with the merge operation.

The main contributions of this paper are fourfold. First, we develop the notions of median membership and imprecision membership to capture the essential information and in maintain consistency of vague data. Second, we define a partial order on merged vague relations which induces a lattice based on O-equivalence classes. We also define a partial order based on the vague values which induces a complete semi-lattice in each O-equivalence class. Third, we extend the satisfaction of an FD in a vague relation in terms values being mi-overlapping rather than equal and show that Lien's and Atzeni's axiom system is sound and complete for FDs being satisfied in vague relations. Finally, we propose the chase procedure for a vague relation r over R, named VChase, as a means of maintaining consistency of r with respect to a set of FDs F. Our main result is that the output $VChase(r, F)$ of the VChase procedure is the most O-precise approximation of r with respect to F.

The rest of the paper is organized as follows. Section 2 presents some basic concepts related to mi-pair, which are used to enhance vague sets and their operations. In Section 3, we discuss the merge operation, based on the less O-precise order. In Section 4, FDs and the $VChase$ procedure of vague relations are introduced. In Section 5, we give a semantic characterization of the $VChase$ procedure of a vague relation, which is also consistent with respect to a set of FDs. Related work is presented in Section 6. And in Section 7, we offer our concluding remarks.

2 Vague Sets and Mi Memberships

In [6, 3, 7], some basic concepts related to the vague relational data model are given. Here we explain how and why the median membership and the imprecision membership are useful to represent vague data. We assume throughout V is a vague set and U is the universe of discourse for V.

2.1 Median Memberships, Imprecision Memberships and Mi-pair Vague Sets

In order to compare vague values, we need to introduce two derived memberships for discussion. The first is called the *median membership*, $M_m = (\alpha + 1 - \beta)/2$, which represents the overall evidence contained in a vague value and is illustrated in Fig. 1.

Definition 1. (Median membership) *The* median membership *of an object $u \in U$ in a vague set V, denoted by $M_m^V(u)$, is defined by $M_m^V(u) = (\alpha(u) + 1 - \beta(u))/2$. Whenever V and u are understood from context, we simply write M_m.*

It can be checked that $0 \leq M_m \leq 1$. In addition, the vague value [1,1] has the highest M_m, which means the corresponding object totally belongs to V (i.e. a crisp value). The vague value [0,0] has the lowest M_m, which informally means that the corresponding object "totally" does not belong to V (i.e. the empty vague value). The higher M_m is, the more crisp the vague value represents.

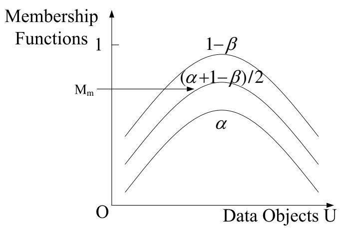

Fig. 1. Median membership of a vague set

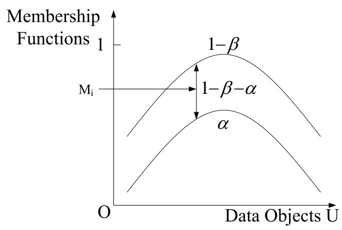

Fig. 2. Imprecision membership of a vague set

The second is called the *imprecision membership*, $M_i = (1 - \beta - \alpha)$, which represents the overall imprecision of a vague value and is illustrated in Fig. 2.

Definition 2. (Imprecision membership) *The* imprecision membership *of an object $u \in U$ in a vague set V, denoted by $M_i^V(u)$, is defined by $M_i^V(u) = 1 - \beta(u) - \alpha(u)$. Whenever V and u are understood from context, we simply write M_i.*

It can be checked that $0 \leq M_i \leq 1$. In addition, the vague value $[a, a] (a \in [0, 1])$ has the lowest M_i which means that we know exactly the membership of the corresponding object (that is, reduced to a fuzzy value). The vague value [0,1] has the highest M_i, which informally means that we know "nothing" about the precision of the corresponding object. The higher M_i is, the more imprecise the vague value represents.

Proposition 1. *The median membership and the imprecision membership of an object satisfy the inequality: $\frac{M_i}{2} \leq M_m \leq (1 - \frac{M_i}{2})$.*

Proposition 1 shows that the median and imprecision memberships actually relate to each other.

Definition 3. **(Mi-pair Vague Set)** *An mi-pair VS vague set, in $U = \{u_1, u_2, \ldots, u_n\}$ is characterized by a median membership function, M_m^V, and an imprecision membership function, M_i^V, where $M_m^V : U \rightarrow [0, 1]$, and $M_i^V : U \rightarrow [0, 1]$. V is given as follows: $V = \sum_{i=1}^{n} < M_m^V(u_i), M_i^V(u_i) > /u_i$. $<M_m^V(u_i), M_i^V(u_i)>/u_i$ is called an element of V and $<M_m^V(u_i), M_i^V(u_i)>$ is called the (mi-pair) vague value of the object u_i.*

Using M_m and M_i, we have a one-to-one correspondence between a vague value, $[\alpha, 1-\beta]$, and *mi-pair* vague value, $<M_m, M_i>$. From now on, a vague set or a vague value refers to an mi-pair vague set or an *mi-pair* vague value, respectively.

Table 4. A sensor vague relation r

	S	**T**	**L**
t_1	<0.7,0.4>/0 + <0.5,1>/3	<0.8,0.3>/0 + <0.6,0.1>/1	<0.4,0.3>/0 + <0.6,0.3>/1
t_2	<0.8,0.1>/0 + <0.1,0.1>/1	<0.9,0.1>/1 + <0.5,0.1>/2	<0.6,0.6>/0 + <0.5,0.2>/2
t_3	<0.9,0.2>/1 + <0.5,0.1>/2	<0.3,0.2>/2	<0.2,0.2>/0
t_4	<0.5,0.1>/3 + <0.8,0.2>/4	<0.4,0.4>/3	<0.4,0.2>/3

Example 1. Let $R = \{S, T, L\}$ be a vague relation schema, where S stands for a sensor ID, T stands for the temperature monitored by a sensor and L stands for a location area ID. A sensor vague relation r having 4 tuples $\{t_1, t_2, t_3, t_4\}$ is shown in Table 4. For those vague elements not listed in the relation, we assume they all have a special vague value $<0, 1>$, which represents the boundary of all vague values, since any median membership is greater than or equal to 0 and any imprecision membership is less than or equal to 1.

2.2 Existence and Overlap of Vague Sets

We next define the concepts of an *mi-existing* VS and overlapping VSs. The underlying idea is to check if vague values satisfy the predefined mi-thresholds: C as a crisp threshold ($0 \leq C \leq 1$), and I as an imprecision threshold ($0 \leq I \leq 1$).

Definition 4. **(Mi-existing)** *Given V and the mi-thresholds C and I, if $\exists u \in U$, such that $M_m^V(u) \geq C$ and $M_i^V(u) \leq I$, then u is an mi-existing object, $<M_m^V(u), M_i^V(u)>/u$ is an mi-existing element, and V is an mi-existing VS under C and I.*

By Definition 4, it follows that V is not mi-existing if all the objects in V are not mi-existing under C and I.

Definition 5. **(Mi-overlap)** *Given two vague sets V_1 and V_2, if $\exists u \in U$, such that $M_m^{V_1}(u) \geq C$ and $M_m^{V_2}(u) \geq C$, $M_i^{V_1}(u) \leq I$ and $M_i^{V_2}(u) \leq I$, then V_1 and V_2 mi-overlap under mi-thresholds C and I, denoted by $V_1 \sim_{mi} V_2(C, I)$. u is called the common mi-existing object of V_1 and V_2 under C and I. Otherwise, V_1 and V_2 do not mi-overlap under C and I, denoted by $V_1 \not\sim_{mi} V_2(C, I)$. We simply write $V_1 \sim_{mi} V_2$ and $V_1 \not\sim_{mi} V_2$, if C and I are known from the context.*

By Definition 5, it follows that V_1 and V_2 do not mi-overlap if there is no common mi-existing object of V_1 and V_2 under C and I.

Example 2. Given C=0.2 and I=0.9, it can be checked that $t_1[L]$ and $t_2[L]$ in Table 4 mi-overlap, i.e. $t_1[L] \sim_{mi} t_2[L](0.2, 0.9)$. However, if C=0.2 and I=0.5, we find that $t_1[L]$ and $t_2[L]$ do not mi-overlap, that is, $t_1[L] \not\sim_{mi} t_2[L](0.2, 0.5)$.

Using the mi-existing objects of VSs, we define mi-union and mi-intersection of VSs.

Definition 6. **(Mi-union)** *Given two vague sets V_1 and V_2 under the mi-thresholds C and I, the mi-union of V_1 and V_2 is a vague set V_3, written as $V_3 = V_1 \vee V_2$, whose median membership and imprecision membership functions are related to those of V_1 and V_2 given as follows. Let $u \in U$.*

1. *If u is an mi-existing object in both V_1 and V_2,*
   $M_m^{V_3}(u) = max(M_m^{V_1}(u), M_m^{V_2}(u))$, $M_i^{V_3}(u) = min(M_i^{V_1}(u), M_i^{V_2}(u))$;
2. *If u is an mi-existing object in V_1 but not in V_2,*
   $M_m^{V_3}(u) = M_m^{V_1}(u)$, $M_i^{V_3}(u) = M_i^{V_1}(u)$;
3. *If u is an mi-existing object in V_2 but not in V_1,*
   $M_m^{V_3}(u) = M_m^{V_2}(u)$, $M_i^{V_3}(u) = M_i^{V_2}(u)$;
4. *If u is not an mi-existing object in both V_1 and V_2,*
   $M_m^{V_3}(u) = M_m^{V_1}(u), M_i^{V_3}(u) = M_i^{V_1}(u)$, *if* $M_m^{V_1}(u) \geq M_m^{V_2}(u)$;
   $M_m^{V_3}(u) = M_m^{V_2}(u), M_i^{V_3}(u) = M_i^{V_2}(u)$, *otherwise.*

Since the fourth case of Def. 6 adopts the vague value from either V_1 or V_2, dependent on which has the higher median membership, it guarantees that the mi-union of two non-mi-existing elements cannot "upgrade" to an mi-existing element. That is to say, it always keeps the elements that do not satisfy mi-thresholds to be non-mi-existing.

Definition 7. **(Mi-intersection)** *Using the same set of notations of Definition 6, the mi-intersection of VSs V_1 and V_2 is a VS V_3, written as $V_3 = V_1 \wedge V_2$, is defined as follows:*

1. *If u is an mi-existing object in both V_1 and V_2,*
   $M_m^{V_3}(u) = max(M_m^{V_1}(u), M_m^{V_2}(u))$, $M_i^{V_3}(u) = min(M_i^{V_1}(u), M_i^{V_2}(u))$;
2. *If u is an mi-existing object in V_1 but not in V_2,*
   $M_m^{V_3}(u) = M_m^{V_2}(u)$, $M_i^{V_3}(u) = M_i^{V_2}(u)$;
3. *If u is an mi-existing object in V_2 but not in V_1,*
   $M_m^{V_3}(u) = M_m^{V_1}(u)$, $M_i^{V_3}(u) = M_i^{V_1}(u)$;
4. *If u is not an mi-existing object in both V_1 and V_2,*
   $M_m^{V_3}(u) = M_m^{V_1}(u), M_i^{V_3}(u) = M_i^{V_1}(u)$, *if* $M_m^{V_1}(u) \geq M_m^{V_2}(u)$;
   $M_m^{V_3}(u) = M_m^{V_2}(u), M_i^{V_3}(u) = M_i^{V_2}(u)$, *otherwise.*

Note that the cases 1 and 4 in Definition 7 are identical to their counterparts in Definition 6.

3 Merge Operation of Vague Relations

In this section, we define the merge of a vague relation r as the operation which replaces each attribute value (represented by a VS) in r by the mi-union of all attribute values with respect to the same reflexive and transitive closure under mi-overlap. This leads to the concept of a less object-precise partial order on merged vague relations.

From now on, we let $R = \{A_1, A_2, \ldots, A_m\}$ be a relation schema and r be a vague relation over R. We also assume common notation used in relational databases [4] such as the projection of a tuple $t[A]$.

The semantics of a vague set, $t[A_i]$, where $t \in r$ and $A_i \in R$, are that an object $u \in U_i$ has the vague value $<M_m(u), M_i(u)>$ in $t[A_i]$. The intuition is that, for those objects which are not mi-existing, we regard their memberships are too weak to consider in the process of chasing the inconsistency with respect to a set of FDs.

We now define the merge operation which replaces each attribute value of a tuple in a vague relation by the mi-union of all attribute values with respect to the same reflexive and transitive closure under mi-overlap.

Definition 8. (Merged relation) *Given $A \in R$ and mi-thresholds C and I, we construct a directed graph $G = (V, E)$, where $V = \pi_A(r)$. An edge ($t_1[A]$, $t_2[A]$) is in E iff $t_1[A] \sim_{mi} t_2[A](C, I)$. Let $G^+ = (V^+, E^+)$ be the reflexive and transitive closure of G. The merge of r, denoted by $merge(r)$, is the vague relation resulting from replacing each $t[A]$ by $\bigvee\{t[A]' | (t[A], t[A]') \in E^+\}$ for all $A \in R$.*

We let $MERGE(R)$ be a collection of all merged relations over R under C and I.

Example 3. Given C=0.2 and I=0.9, the vague relation $merge(r)$, is shown in Table 5, where r is shown in Table 4. For example, since $t_1[L] \sim_{mi} t_2[L](0.2, 0.9)$ and $t_2[L] \sim_{mi} t_3[L](0.2, 0.9)$, we replace $t_1[L]$, $t_2[L]$ and $t_3[L]$ by <0.6,0.2>/0 + <0.6,0.3>/1 + <0.5,0.2>/2. Note that the first two tuples in r (t_1 and t_2) have been merged into a single tuple (t'_1) in $merge(r)$. With different mi-thresholds C and I, we may have different merge results. If we set C=0.2 and I=0.5, then $t_1[L] \not\sim_{mi} t_2[L](0.2, 0.5)$. In this case, we obtain $merge(r)$ shown in Table 6. We see that the first two tuples (t'_1 and t'_2) are not merged.

Table 5. A relation $merge(r)$ under $C = 0.2$ and $I = 0.9$

	S	**T**	**L**
t'_1	<0.8,0.1>/0 + <0.1,0.1>/1 + <0.5,1>/3	<0.8,0.3>/0 + <0.9,0.1>/1 + <0.5,0.1>/2	<0.6,0.2>/0 + <0.6,0.3>/1 + <0.5,0.2>/2
t'_2	<0.9,0.2>/1 + <0.5,0.1>/2	<0.8,0.3>/0 + <0.9,0.1>/1 + <0.5,0.1>/2	<0.6,0.2>/0 + <0.6,0.3>/1 + <0.5,0.2>/2
t'_3	<0.5,0.1>/3 + <0.8,0.2>/4	<0.4,0.4>/3	<0.4,0.2>/3

Table 6. A relation $merge(r)$ under $C = 0.2$ and $I = 0.5$

	S	**T**	**L**
t_1'	<0.8,0.1>/0 + <0.1,0.1>/1 + <0.5,1>/3	<0.8,0.3>/0 + <0.9,0.1>/1 + <0.5,0.1>/2	<0.4,0.2>/0 + <0.6,0.3>/1
t_2'	<0.8,0.1>/0 + <0.1,0.1>/1 + <0.5,1>/3	<0.8,0.3>/0 + <0.9,0.1>/1 + <0.5,0.1>/2	<0.6,0.6>/0 + <0.5,0.2>/2
t_3'	<0.9,0.2>/1 + <0.5,0.1>/2	<0.8,0.3>/0 + <0.9,0.1>/1 + <0.5,0.1>/2	<0.4,0.2>/0 + <0.6,0.3>/1
t_4'	<0.5,0.1>/3 + <0.8,0.2>/4	<0.4,0.4>/3	<0.4,0.2>/3

There are two levels of precision we consider in vague sets for handling inconsistency. The first is the *object-precision*, which intuitively means the precision according to the cardinality of a set of mi-existing objects. The second is, given the same object, the vague values have different mi precision, which we term the *value-precision*.

We first define a partial order named *less* $object$*-precise* on VSs based on mi-existing objects and extend this partial order to tuples and relations in $MERGE(R)$.

Definition 9. (Less object-precise and object-equivalence) *We define a partial order, less* $object$*-precise (or less O-precise for simplicity) between two vague sets* V_1 *and* V_2 *as follows:*

$V_1 \sqsubseteq_O V_2$ *if the set of* mi*-existing objects in* V_1 *is a superset of the set of those in* V_2*. We say that* V_1 *is less O-precise than* V_2*.*

We extend $\sqsubseteq_O$ *in* r *as follows. Let* t_1, $t_2 \in r$. $t_1 \sqsubseteq_O t_2$ *if* $\forall A_i \in R$, $t_1[A_i] \sqsubseteq_O t_2[A_i]$*. We say that* t_1 *is less O-precise than* t_2*.*

Finally, we extend $\sqsubseteq_O$ *in* $MERGE(R)$ *as follows: Let* $r_1, r_2 \in MERGE(R)$. $r_1 \sqsubseteq_O r_2$ *if* $\forall t_2 \in r_2$, $\exists t_1 \in r_1$ *such that* $t_1 \sqsubseteq_O t_2$*. We say that* r_1 *is less O-precise than* r_2*.*

We define an object-equivalence between V_1 *and* V_2*, denoted as* $V_1 \doteq_O V_2$*, iff* $V_1 \sqsubseteq_O V_2$ *and* $V_2 \sqsubseteq_O V_1$*. Similar definitions of object-equivalence are extended to tuples and relations.*

Thus, an object-equivalence relation on $MERGE(R)$ induces a partition of $MERGE(R)$, which means all vague relations equivalent to each other are put into one *O-equivalence class*. Given any two vague relations in an O-equivalence class of $MERGE(R)$, each tuple in one vague relation has a one-to-one correspondence in the other vague relation. With in an O-equivalence class of $MERGE(R)$, we still have to consider the second level of precision as follows:

Definition 10. (Less value-precise and value-equivalence) *Let* $V_1 \doteq_O V_2$*. We define a partial order, less* $value$*-precise (or less* V*-precise for simplicity), between* V_1 *and* V_2 *as follows:*

Let $a = <M_m^{V_1}, M_i^{V_1}>$ *and* $b = < M_m^{V_2}, M_i^{V_2} >$ *be the respective vague values of a common* mi*-existing object* u *in* V_1 *and* V_2*. If* $M_m^{V_1} \leq M_m^{V_2}$ *and* $M_i^{V_1} \geq M_i^{V_2}$ *(that is,* a *is less crisp and more imprecise than* b*), then we say* a *is less* V*-precise than* b*, denoted as* $a \sqsubseteq_V b$*.*

$V_1 \sqsubseteq_V V_2$ if the vague value of each mi-existing object in V_1 is less V-precise than that of the same object in V_2. We say that V_1 is less V-precise than V_2.

We extend $\sqsubseteq_V$ in r as follows. Let t_1, $t_2 \in r$ and $t_1 \doteq_O t_2$. $t_1 \sqsubseteq_V t_2$ if $\forall A_i \in R$, $t_1[A_i] \sqsubseteq_V t_2[A_i]$. We say that t_1 is less V-precise than t_2.

Finally, we extend $\sqsubseteq_V$ in an O-equivalence class of $MERGE(R)$ as follows. Let $r_1 \doteq_O r_2$. $r_1 \sqsubseteq_V r_2$ if $\forall t_1 \in r_1$, $\exists t_2 \in r_2$ such that $t_1 \sqsubseteq_V t_2$. We say that r_1 is less V-precise than r_2.

We define a value-equivalence, denoted as $V_1 \doteq_V V_2$ iff $V_1 \sqsubseteq_V V_2$ and $V_2 \sqsubseteq_V V_1$. Similar definitions are extended to tuples and relations.

According to Definition 10, we define V-join $\cup$ and V-meet $\cap$ under $\sqsubseteq_V$ of vague values of a given object, that is, $< M_m^x, M_i^x > \cup < M_m^y, M_i^y > = < max\{M_m^x, M_m^y\}, min\{M_i^x, M_i^x\} >$ and $< M_m^x, M_i^x > \cap < M_m^y, M_i^y > = < min\{M_m^x, M_m^y\}, max\{M_i^x, M_i^x\} >$. It is easy to check that the less V-precise order $\sqsubseteq_V$ induces a complete semi-lattice by using $\cup$ and $\cap$ as shown in Fig. 3.

It can be checked that <1,0> is the top element according to the less V-precise order. Note that for some mi-pair vague values, V-meet may cause the corresponding vague value $[\alpha(u), 1-\beta(u)]$ beyond the legal range [0,1], which is not valid. From now on, we restrict our discussion to the V-meet that gives rise to valid vague values as a result.

Given any mi-thresholds C and I, if $<C, I>$ is a valid vague value, then we can use $<C, I>$ as a cut-off boundary to construct a complete lattice, rather than the original complete semi-lattice shown in Fig. 3, induced by the less V-precise order $\sqsubseteq_V$. For example, given $<C, I>$ = $<0.5, 0.5>$ (or $<0.6, 0.4>$), which is a valid vague value, in the dotted-line region in Fig. 3, all vague values form a complete lattice, since given any two values in the enclosed region, we have their greatest lower bound and lowest upper bound. However, if $<C, I>$ is not a valid vague value, then we have a complete semi-lattice, since some values in the enclosed region constructed by $<C, I>$ do not have their greatest lower bound. For instance, in the dotted-line region with respect to an invalid vague value $<0.1, 0.3>$, all vague values form a complete semi-lattice, since for $<0.1, 0.2>$ and $<0.2, 0.3>$, we do not have their greatest lower bound.

From Definition 9, we can deduce that $MERGE(R)$ is a lattice based on O-equivalence classes with respect to $\sqsubseteq_O$. In this lattice, each node is an O-equivalence class, in which all vague relations are O-equivalent. The top node is the O-equivalence class of $\emptyset^O$, i.e. the set of vague relations with an empty set of tuples. The bottom node is the O-equivalence class, in which all vague relations have only one tuple and all mi-existing objects in vague relations form the universes of discourse.

Example 4. For simplicity we just assume $U = \{0, 1\}$ and $R = A$, we construct the lattice for $MERGE(R)$ under C=0.5 and I=0.5 according to O-equivalence classes. As shown in Fig. 4, all O-equivalence classes (the nodes represented by circles) form a lattice based on $\sqsubseteq_O$. Each node in the lattice is actually the set of all vague relations (represented by tables with single attribute) which are O-equivalent to each other. For instance, r_1 and r_2 are two vague relations with two tuples such that $r_1 \doteq_O r_2$. Similarly, we have $r_3 \doteq_O r_4$, where r_3 and r_4 are two vague relations with only one tuple. Inside each node, based on $\sqsubseteq_V$ in Definition 10, all vague relations in the node form

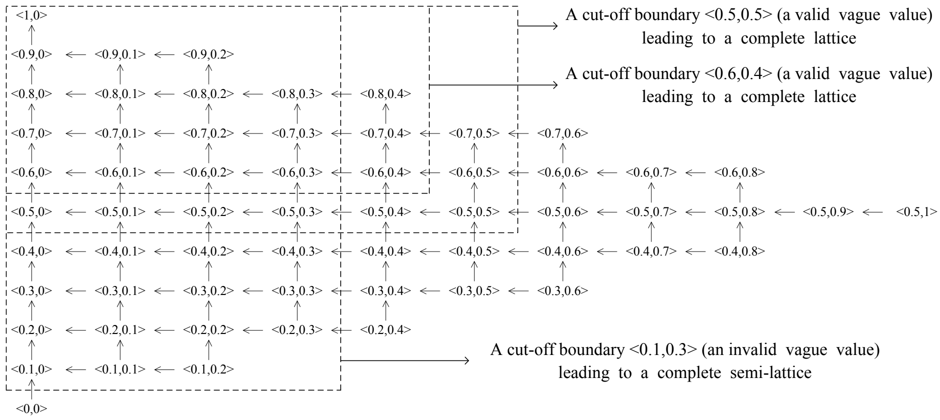

Fig. 3. A complete semi-lattice of vague values of an object u

a complete lattice (when the cut-off boundary is a valid vague value) or a complete semi-lattice (when the cut-off boundary is not a valid vague value). In the complete (semi-)lattice, the top element is the vague relation in which all vague values of objects are $<$1,0$>$, and if the cut-off boundary $<C,I>$ is a valid vague value, then the bottom element is the vague relation in which all vague values of objects are $<C,I>$. For example, in the lattice shown in Fig. 6, which is the bottom node of the lattice in Fig. 4, each table represents a single attribute vague relation. The top is the single attribute vague relation r_t with one tuple $<<$1,0$>$/0+$<$1,0$>$/1$>$. The bottom is the vague relation r_b with single tuple $<<$0.5,0.5$>$/0+$<$0.5,0.5$>$/1$>$, and the vague value of each object is $<C,I>$.

Given different mi-thresholds, a lattice induced by $\sqsubseteq_V$ exists inside each node. For instance, we have the lattice of $MERGE(R)$ under C=0.5 and I=0.4 as shown in Fig. 5. The bottom elements in each node are different from those in Fig. 4, since the mi-thresholds are different.

Now, we extend the mi-existing of VSs given in Definition 4 to tuples as follows: $t[X]$ is mi-existing, if $\forall A \in X$, $t[A]$ is mi-existing, where $X \subseteq R$. We also extend the concept of mi-overlap given in Definition 5 to tuples $t_1, t_2 \in r$ under mi-thresholds C and I as follows: $t_1[X] \sim_{mi} t_2[X](C,I)$, if $\forall A \in X$, $t_1[A] \sim_{mi} t_2[A](C,I)$ where $X \subseteq R$.

Example 5. We can verify that $t_1 \sim_{mi} t_2(0.2, 0.9)$ in the relation shown in Table 4.

4 Functional Dependencies and Vague Chase

Functional Dependencies (FDs) being satisfied in a vague relation r can be formalized in terms values being mi-overlapping rather than equal. The VChase procedure for r is a means of maintaining consistency of r with respect to a given set of FDs.

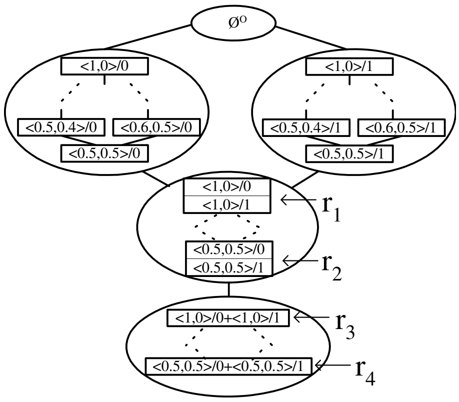

Fig. 4. A lattice of $MERGE(R)$ under C=0.5 and I=0.5

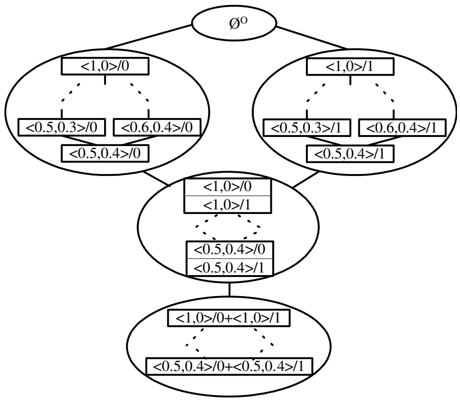

Fig. 5. A lattice of $MERGE(R)$ under C=0.5 and I=0.4

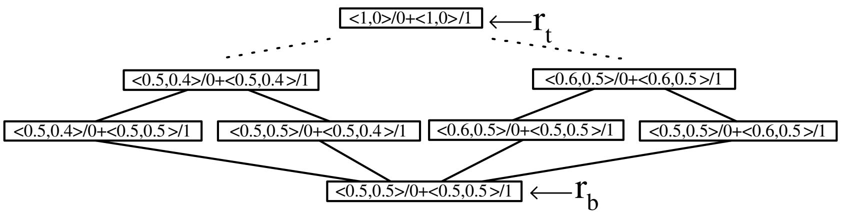

Fig. 6. A lattice within the bottom node of the lattice of MERGE(R) of Fig. 4

4.1 Functional Dependencies in Vague Relations

We formalize the notion of an FD being satisfied in a vague relation. Lien's and Atzeni's axiom system is sound and complete for FDs being satisfied in vague relations.

Definition 11. **(Functional dependency)** *Given* mi*-thresholds* C *and* I*, a* Functional Dependency *over* R *(or simply an FD) is a statement of the form* $X \rightarrow_{C,I} Y$*, where* $X, Y \subseteq R$*. We may simply write* $X \rightarrow Y$ *if* C *and* I *are known from context. An FD* $X \rightarrow Y$ *is satisfied in a relation* r*, denoted by* $r \models X \rightarrow Y$*, if* $\forall t_1, t_2 \in r$*,* $t_1[X] \sim_{mi} t_2[X](C, I)$*, then* $t_1[Y] \sim_{mi} t_2[Y](C, I)$*, or* $t_1[Y]$ *or* $t_2[Y]$ *are not* mi*-existing.*

A set of FDs F over R is satisfied in r, denoted by $r \models F$, if $\forall X \rightarrow Y \in F$, $r \models X \rightarrow Y$. If $r \models F$ we say that r is *consistent* with respect to F (or simply r is consistent if F is understood from context); otherwise if $r \not\models F$ then we say that r is *inconsistent* with respect to F (or simply r is inconsistent). We let $SAT(F)$ denote the finite set $\{r \in MERGE(R) | r \models F\}$.

Example 6. Let $F = \{S \rightarrow_{0.2,0.9} TL, L \rightarrow_{0.2,0.9} S\}$ be a set of FDs over R, where R is the relation schema whose semantics are given in Example 1. We can verify that

$r \vDash S \rightarrow_{0.2,0.9} TL$ but that $r \nvDash L \rightarrow_{0.2,0.9} S$, where r is the relation shown in Table 4. Thus $r \in SAT(\{S \rightarrow_{0.2,0.9} TL\})$ but $r \notin SAT(F)$. Consider also $merge(r)$ shown in Table 5, we have $merge(r) \in SAT(\{S \rightarrow_{0.2,0.9} TL\})$ but $merge(r) \notin SAT(F)$. If we change the mi-thresholds from 0.2 and 0.9 to 0.2 and 0.5, the result is different. Let $F = \{S \rightarrow_{0.2,0.5} TL, L \rightarrow_{0.2,0.5} S\}$ be a set of FDs over R. We can verify that $r \nvDash S \rightarrow_{0.2,0.5} TL$ and that $r \nvDash L \rightarrow_{0.2,0.5} S$. Thus $r \notin SAT(\{S \rightarrow_{0.2,0.5} TL\})$ and $r \notin SAT(F)$. Consider also $merge(r)$ shown in Table 6, we have $merge(r) \notin SAT(\{S \rightarrow_{0.2,0.5} TL\})$ and $merge(r) \notin SAT(F)$.

We say that F logically implies an FD $X \rightarrow_{C,I} Y$ over R written $F \vDash X \rightarrow_{C,I} Y$, whenever for any domain D, $\forall r \in REL_D(R)$, if $r \vDash F$ holds then $r \vDash X \rightarrow Y$ also holds.

Here we state the well known Lien's and Atzeni's axiom system [5, 4] for incomplete relations as follows:

1. *Reflexivity*: If $Y \subseteq X$, then $F \vdash X \rightarrow Y$.
2. *Augmentation*: If $F \vdash X \rightarrow Y$ holds, then $F \vdash XZ \rightarrow YZ$ also holds.
3. *Union*: If $F \vdash X \rightarrow Y$ and $F \vdash X \rightarrow Z$ hold, then $F \vdash X \rightarrow YZ$ holds.
4. *Decomposition*: If $F \vdash X \rightarrow YZ$ holds, then $F \vdash X \rightarrow Y$ and $F \vdash X \rightarrow Z$ hold.

Definition 12. (Soundness and Completeness of Axiom system) *Whenever an FD $X \rightarrow Y$ can be proven from F using a finite number of inference rules from Lien's and Atzeni's axiom system [4], we write $F \vdash X \rightarrow Y$.*

Lien's and Atzeni's axiom system is sound if $F \vdash X \rightarrow Y$ implies $F \vDash X \rightarrow Y$. Correspondingly, Lien's and Atzeni's axiom system is complete if $F \vDash X \rightarrow Y$ implies $F \vdash X \rightarrow Y$.

The proof of the following theorem is standard [4], which we establish a counter example relation to show that $F \nvdash X \rightarrow Y$ but $F \nvDash X \rightarrow Y$. Due to lack of space, we omit all proofs in this paper. However, all proofs will be contained in the full version of it.

Theorem 1. *Lien's and Atzeni's axiom system is sound and complete for FDs being satisfied in vague relations.*

4.2 Vague Chase

We define the chase procedure for maintaining consistency in vague relations. Assuming that a vague relation r is updated with information obtained from several different sources, at any given time the vague relation r may be inconsistent with respect to a set of FDs F. Thus we input r and F into the VChase procedure and its output, denoted by $VChase(r, F)$, is a consistent relation over R with respect to F. The pseudo-code for the algorithm $VChase(r, F)$ is presented in Algorithm 1.

We call an execution of line 6 in Algorithm 1 a *VChase step*, and say that the VChase step *applies* the FD $X \rightarrow Y$ to the *current state* of $VChase(r, F)$.

Algorithm 1 $VChase(r, F)$

```
Result := r;
Tmp := ∅;
while Tmp ≠ Result do
    Tmp := Result;
    if X →_{C,I} Y ∈ F, ∃t1, t2 ∈ Result such that t1[X] ∼_mi t2[X](C, I), t1[Y] and t2[Y]
    are mi-existing but t1[Y] ≁_mi t2[Y](C, I) then
        ∀A ∈ (Y − X), t1[A], t2[A]:=t1[A] ∨ t2[A];
    end if
end while
return merge(Result);
```

Table 7. Vague relation $VChase(r, F)$ under C=0.2 and I=0.9

S	**T**	**L**
<0.8,0.1>/0 + <0.9,0.2>/1 + <0.5,0.1>/2 + <0.5,1>/3	<0.8,0.3>/0 + <0.9,0.1>/1 + <0.5,0.1>/2	<0.6,0.2>/0 + <0.6,0.3>/1 + <0.5,0.2>/2
<0.5,0.1>/3 + <0.8,0.2>/4	<0.4,0.4>/3	<0.4,0.2>/3

Table 8. Vague relation $VChase(r, F)$ under C=0.2 and I=0.5

S	**T**	**L**
<0.8,0.1>/0 + <0.9,0.2>/1 + <0.5,0.1>/2 + <0.5,1>/3	<0.8,0.3>/0 + <0.9,0.1>/1 + <0.5,0.1>/2	<0.4,0.2>/0 + <0.6,0.3>/1 + <0.5,0.2>/2
<0.5,0.1>/3 + <0.8,0.2>/4	<0.4,0.4>/3	<0.4,0.2>/3

Example 7. The vague relation $VChase(r, F)$ is shown in Table 7, where r is shown in Table 4 and $F = \{S \rightarrow_{0.2,0.9} TL, L \rightarrow_{0.2,0.9} S\}$ is the set of FDs over R. We can verify that $VChase(r, F) \models F$, i.e. $VChase(r, F)$ is consistent, and that $VChase(r, F) = VChase(merge(r), F)$, where $merge(r)$ is shown in Table 5. If $F = \{S \rightarrow_{0.2,0.5} TL, L \rightarrow_{0.2,0.5} S\}$ is the set of FDs over R, we can also verify that $VChase(r, F) \models F$, which is as shown in Table 8, and that $VChase(r, F) = VChase(merge(r), F)$, where $merge(r)$ is shown in Table 6.

From Tables 7 and 8, we see that different mi-thresholds C and I may give rise to different VChase results (the corresponding values of L in the first tuple).

The next lemma shows that $VChase(r, F)$ is less O-precise than $merge(r)$ and unique. Its complexity is polynomial time in the sizes of r and F.

Lemma 1. *The following statements are true:*

1. *VChase(r, F)* $\sqsubseteq_O$ *merge(r).*
2. *VChase(r, F) is unique.*
3. *VChase(r, F) terminates in polynomial time in the sizes of* r *and* F.

The next theorem shows that the VChase procedure outputs a consistent relation and that it commutes with the merge operation.

Theorem 2. *The following two statements are true:*
1. VChase(r, F) $\models$ F, i.e. VChase(r, F) is consistent.
2. VChase(r, F) = VChase(merge(r), F).

5 The Most O-precise Approximation of a Vague Relation

The $VChase(r, F)$ procedure can be regarded as the most O-precise approximation of r, which is also consistent to F. In this section, we first define the *join* of vague relations, which corresponds to the least upper bound of these relations in the lattice $MERGE(R)$ based on O-equivalence classes. (Recall the lattices shown in Figures 4 and 5.) Next, we define the *most O-precise approximation* of r with respect to F to be the *join* of all the consistent and merged relations which are less O-precise than r. Our main result is that $VChase(r, F)$ is the most O-precise approximation of r with respect to F. Thus, the VChase procedure solves the consistency problem in polynomial time in the size of r and F.

We now define the join operation on relations in the lattice of $MERGE(R)$ based on O-equivalence classes.

Definition 13. (Join operation) *The join of two vague relations, $r_1, r_2 \in MERGE(R)$, denoted by $r_1 \sqcup r_2$, is given by*
$r_1 \sqcup r_2 = \{t | \exists t_1 \in r_1, \exists t_2 \in r_2$ *such that* $\forall A \in R$, $t_1[A] \sim_{mi} t_2[A](C, I)$, $t[A] = t_1[A] \wedge t_2[A]\}$.

It can be verified that the O-equivalence class that consists of $r_1 \sqcup r_2$ is the least upper bound with respect to the O-equivalence classes of r_1 and r_2 in $MERGE(R)$. From now on we will assume that $r_1, r_2 \in MERGE(R)$.

The next theorem shows that if two relations are consistent then their join is also consistent.

Theorem 3. *Let $r_1, r_2 \in SAT(F)$. Then $r_1 \sqcup r_2 \in SAT(F)$.*

The most O-precise approximation of a vague relation r over R with respect to F is the join of all consistent relations s such that s is a merged relation that is less O-precise than r.

Definition 14. (Most O-precise approximation) *The most O-precise approximation of a vague relation r with respect to F, denoted by $approx(r, F)$, is given by $\bigsqcup\{s | s \sqsubseteq_O merge(r)$ and $s \in SAT(F)\}$.*

The next lemma shows some desirable properties of approximations.

Lemma 2. *The following statements are true:*

1. *approx(r, F) is consistent.*
2. *approx(r, F) $\sqsubseteq_O$ merge(r).*
3. *approx(r, F) $\doteq_O$ merge(r) iff r is consistent.*

The next theorem, which is the main result of this section, shows that output of the VChase procedure is equal to the corresponding most O-precise approximation. Thus, the vague relation $VChase(r, F)$, which is shown in Table 7, is the most O-precise approximation of r with respect to F, where r is the relation over R shown in Table 4 and F is the set of FDs over R specified in Example 6.

Theorem 4. *$VChase(r, F) \doteq_O approx(r, F)$.*

6 Related Work

The problem of maintaining the consistency with respect to FDs of a relational database is well-known. However, in many real applications, it is too restrictive for us to have FDs hold in relations. For example, the salary of employees is approximately determined by the number of working years. The discovery of meaningful but approximate FDs is an interesting topic in both data mining and database areas [8]. Thus, many research works on approximate FDs have been proposed [9–11]. In order to deal with uncertain information including missing, unknown, or imprecisely known data, probability theory [12–15], fuzzy set and possibility theory-based treatments [16, 17] have been applied to extend standard relational databases and FDs [18–22]. Based on vague set theory, we apply some useful parameters such as the median and imprecision memberships to characterize uncertain data objects. The parameters are used to extend various concepts such as satisfaction of FDs in vague relations.

The work in [23] introduces the notion of imprecise relations and FDs being satisfied in imprecise relations in order to cater for the situation when the information may be obtained from different sources and therefore may be imprecise. However, we apply the interval-based vague memberships, which capture positive, neutral and negative information of objects, and extend the "equally likely objects" assumption used in [23]. The imprecise set in [23] can also be considered as the O-equivalent VS in our work.

7 Conclusions

In this paper, we extend FDs to be satisfied in a vague relation. We define the mi-overlap between vague sets and the merge operation of a vague relation r which replaces each attribute value in r by the mi-union of all attribute values with respect to the same reflexive and transitive closure under mi-overlap. We also define a partial order on merged vague relations which induces a lattice on the set of merged vague relations based on O-equivalence classes. Inside each O-equivalence class, we define a partial order based on the vague values of mi-existing objects which induces a complete semi-lattice. Satisfaction of an FD in a vague relation is defined in terms values being mi-overlapping rather than equality. Lien's and Atzeni's axiom system is sound and complete for FDs being satisfied in vague relations. We define the chase procedure VChase as a means of maintaining consistency of r with respect to F. Our main result is that VChase outputs the most O-precise approximation of r with respect to F and can be computed in polynomial time in the sizes of r and F. Our result suggests a mechanical way that maintains the consistency of vague data. It is both interesting and challenging to use the VChase result to provide more effective and efficient evaluation of SQL over vague relations as a future work.

References

1. Zadeh, L.A.: Fuzzy sets. Information and Control **8** (1965) 338–353
2. Gau, W.L., Buehrer, D.J.: Vague sets. IEEE Transactions on Systems, Man, and Cybernetics **23** (1993) 610–614
3. Lu, A., Ng, W.: Vague sets or intuitionistic fuzzy sets for handling vague data: Which one is better? In: ER. (2005) 401–416
4. Atzeni, P., Antonellis, V.D.: Relational Database Theory. Benjamin/Cummings (1993)
5. Lien, Y.E.: On the equivalence of database models. J. ACM **29** (1982) 333–362
6. Lu, A., Ng, W.: Managing merged data by vague functional dependencies. In: ER. (2004) 259–272
7. Lu, A., Ng, W.: Mining hesitation information by vague association rules. In: ER. (2007)
8. Bra, P.D., Paredaens, J.: Horizontal decompositions for handling exceptions to functional dependencies. In: Advances in Data Base Theory. (1982) 123–141
9. Kivinen, J., Mannila, H.: Approximate inference of functional dependencies from relations. Theor. Comput. Sci. **149** (1995) 129–149
10. Huhtala, Y., Kärkkäinen, J., Porkka, P., Toivonen, H.: Tane: An efficient algorithm for discovering functional and approximate dependencies. Comput. J. **42** (1999) 100–111
11. King, R.S., Legendre, J.J.: Discovery of functional and approximate functional dependencies in relational databases. JAMDS **7** (2003) 49–59
12. Barbará, D., Garcia-Molina, H., Porter, D.: The management of probabilistic data. IEEE Trans. Knowl. Data Eng. **4** (1992) 487–502
13. Dey, D., Sarkar, S.: A probabilistic relational model and algebra. ACM Trans. Database Syst. **21** (1996) 339–369
14. Ross, R., Subrahmanian, V.S., Grant, J.: Aggregate operators in probabilistic databases. J. ACM **52** (2005) 54–101
15. Lakshmanan, L.V.S., Leone, N., Ross, R., Subrahmanian, V.S.: Probview: A flexible probabilistic database system. ACM Trans. Database Syst. **22** (1997) 419–469
16. Bosc, P., Prade, H.: An introduction to the fuzzy set and possibility theory-based treatment of flexible queries and uncertain or imprecise databases. In Motro, A., Smets, P., eds.: Uncertainty Management in Information Systems: From Needs to Solutions. Kluwer Academic Publ. (1996) 285–324
17. Dubois, D., Prade, H.: Possibility Theory: An Approach to Computerized Processing of Uncertainty. Plenum Press, New York (1988)
18. Dey, D., Sarkar, S.: Generalized normal forms for probabilistic relational data. IEEE Trans. Knowl. Data Eng. **14** (2002) 485–497
19. Raju, K.V.S.V.N., Majumdar, A.K.: Fuzzy functional dependencies and lossless join decomposition of fuzzy relational database systems. ACM Trans. Database Syst. **13** (1988) 129–166
20. Bosc, P., Dubois, D., Prade, H.: Fuzzy functional dependencies and redundancy elimination. JASIS **49** (1998) 217–235
21. Intan, R., Mukaidono, M.: Fuzzy functional dependency and its application to approximate data querying. In Desai, B.C., Kiyoki, Y., Toyama, M., eds.: IDEAS, IEEE Computer Society (2000) 47–54
22. Brown, P., Haas, P.J.: Bhunt: Automatic discovery of fuzzy algebraic constraints in relational data. In: VLDB. (2003) 668–679
23. Levene, M.: Maintaining consistency of imprecise relations. Comput. J. **39** (1996) 114–123