Fuzzy Functional Dependencies and Lossless Join Decomposition of Fuzzy Relational Database Systems

K. V. S. V. N. RAJU
Andhra University
and
ARUN K. MAJUMDAR
University of Guelph

This paper deals with the application of fuzzy logic in a relational database environment with the objective of capturing more meaning of the data. It is shown that with suitable interpretations for the fuzzy membership functions, a fuzzy relational data model can be used to represent ambiguities in data values as well as impreciseness in the association among them. Relational operators for fuzzy relations have been studied, and applicability of fuzzy logic in capturing integrity constraints has been investigated. By introducing a fuzzy resemblance measure EQUAL for comparing domain values, the definition of classical functional dependency has been generalized to fuzzy functional dependency (**ffd**). The implication problem of **ffds** has been examined and a set of sound and complete inference axioms has been proposed. Next, the problem of lossless join decomposition of fuzzy relations for a given set of fuzzy functional dependencies is investigated. It is proved that with a suitable restriction on EQUAL, the design theory of a classical relational database with functional dependencies can be extended to fuzzy relations satisfying fuzzy functional dependencies.

Categories and Subject Descriptors: E.5 [**Data**]: Files; H.2.1 [**Database Management**]: Logical Design; I.5.1 [**Pattern Recognition**]: Models

General Terms: Design, Theory

Additional Key Words and Phrases: Functional dependency, fuzzy functional dependency, fuzzy logic, fuzzy measure, fuzzy relation, inference axiom, lossless join, membership function, relational database

1. INTRODUCTION

Since Codd [16, 17] proposed the relational data model, relational database systems have been extensively studied and several commercial relational database systems are currently available [19, 29, 45, 46]. This data model usually takes care of only well-defined and unambiguous data. However, in real world

Authors' addresses: K. V. S. V. N. Raju, Department of Computer Science and Systems Engineering, Andhra University College of Engineering, Visakhapatnam 530 003, India. Arun K. Majumdar, Department of Computing and Information Science, University of Guelph, Ontario, Canada N1G 2W1. (On leave from the Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur 721302, India.)

applications data are often partially known (incomplete) or imprecise. For example, instead of specifying that the height of John is 192 cm, one may say that the height of John is around 190 cm, or simply that John is tall. Such statements contain information about the height of John and may be useful in answering queries or making inferences based on age.

In order to capture more meaning of the data, several extensions of the classical relational model have been proposed [7, 8, 18, 27, 28, 29]. In some of these extensions, a variety of "null values" have been introduced to model unknown or not-applicable data values. Attempts have also been made to generalize operators of relational algebra to manipulate such extended data models [7, 18, 29]. As an alternative approach, Rieter [39] has suggested the usage of first order predicate calculus where Skolem functions are used to represent "null values." In the quest for capturing more meaning of the data, it would be natural to extend the classical relational model where partially known (as well as imprecise/fuzzy) data values, such as "salary of John is around $60,000" or "John has a high salary," are permitted. The fuzzy set theory and fuzzy logic proposed by Zadeh [50–56] provide a requisite mathematical framework for dealing with such extended data values. Recently, some authors [2, 3, 9–15, 25, 33–36, 42, 47, 48, 57, 58] have studied relational databases in the light of fuzzy set theory with an objective to accommodate a wider range of real-world requirements and to provide closer man–machine interactions. In some of these proposals, the classical relational algebra operations such as join, projection, etc., have been appropriately extended and the problems related to query language design and query evaluation have been examined.

As we extend the classical relational data model to deal with fuzzy information, it would be necessary to consider integrity constraints that may involve fuzzy concepts. In fact, fuzzy integrity constraints, such as "salaries of almost equally qualified employees should be more or less equal," will arise naturally in fuzzy databases. In classical relational database literature, integrity constraints and associated inference rules constitute a major area of research. Different types of integrity constraints, such as functional dependency, multivalued dependency, join dependency, etc., have been identified and sets of sound and complete inference rules for such dependencies have been proposed [4–6, 19, 21, 22, 24, 29, 30, 43, 46]. Several algorithms have been suggested to design normalized database schemes from a given set of data dependencies [4, 19, 21, 22, 29, 46]. It can also be ensured that the selected database scheme enjoys the lossless join property [1, 29, 30, 46].

To deal with fuzzy data constraints, Zadeh [53, 54] has introduced the concept of particularization (restriction) of a fuzzy relation due to a fuzzy proposition. Just as well-formed formulas of first order calculus can be used to represent integrity constraints in a classical relational database [29, 39, 46], fuzzy integrity constraints can be represented by suitable fuzzy propositions. Moreover, the particularization of a fuzzy relational database due to a set of fuzzy integrity constraints can be computed by combining the fuzzy propositions associated with these integrity constraints according to the rules of fuzzy calculus. In this paper we examine different types of fuzzy relations and particularizations of such relations due to fuzzy integrity constraints. Our primary objective is to extend the design theory of relational databases to the fuzzy domain by suitably defining the fuzzy functional dependency (**ffd**). A set of sound and complete inference rules for fuzzy functional dependencies is proposed and the lossless join problem of fuzzy relations is examined. In view of this, the paper is organized as follows. Section 2 of the paper deals with some of the basic definitions and concepts of classical relational database theory. In Sections 3 and 4 we introduce a few basic definitions from fuzzy set theory and discuss fuzzy relations and fuzzy integrity constraints. In Section 5, fuzzy functional dependency and associated inference rules are examined. Section 6 is concerned with the lossless join of fuzzy relations in the presence of fuzzy functional dependencies.

2. THE CLASSICAL RELATIONAL MODEL

In this section some basic definitions and concepts related to the classical relational data model [19, 29, 46] are given.

Attributes are symbols taken from a finite set $U = \{A_1, A_2, \ldots, A_n\}$. Each attribute A_i has associated with it a domain denoted by $\mathrm{dom}(A_i)$, which is the set of possible values for that attribute. Elements of dom(A), dom(B), and dom(C) are usually denoted by a, b, c, respectively, with possible suffixes. For a set of attributes X, an X-value is an assignment of values to the attributes of X from their domain. We will use the letters A, B, . . . for single attributes and the letters X, Y, . . . for sets of attributes. The union of two sets X and Y is written as XY. Also no distinction is made between a single attribute and the set {A}.

A relation r with attributes $\{A_1, A_2, \ldots, A_n\}$ is a subset of the Cartesian product of $\mathrm{dom}(A_1) \times \cdots \times \mathrm{dom}(A_n)$. A relation scheme on $\{A_1, \ldots, A_n\}$ will be denoted by $R(A_1 \ldots A_n)$ or R. A relation r is considered to be an instance of a relation scheme R. The elements of the relation are called tuples or rows. A tuple is usually represented as a string of values associated with the attributes; for example, **ac** is a tuple of a relation r on R(AC). If t is a tuple in a relation r of R(U), and A is an attribute in U, then t[A] is the A-component of t. Similarly for the set of attributes $X \subseteq U$, we use the notation t[X] to denote the restriction of t to X; for example if t = abc, then t[AC] = ac.

There are two operations on relations that are of interest to us: *projection* and *natural join.* The projection of a relation r of R(XYZ) over the set of attributes X is obtained by taking the restriction of the tuples of r to the attributes in X and eliminating duplicate tuples in what remains. This operation is usually denoted by

$$\Pi_X(r) = \{t[X] \mid t \in r\} \tag{2.1}$$

Since in the fuzzy set literature the symbol Π is used to denote the possibility distribution, in this paper we will use the notation $\mathbf{P}_X(r)$ for the projection of r on X.

Let r_1 and r_2 be two relations of R(XY) and R(XZ), respectively. The natural join $r_1 \bowtie r_2$ is a relation over R(XYZ) defined by

$$r = r_1 \bowtie r_2 = \{t \mid t[XY] \in r_1 \text{ and } t[YZ] \in r_2\} \tag{2.2}$$

Among the different types of data dependencies that have been identified so far, the functional dependency requires special mention. In fact, the importance of the functional dependency in the design of relational database systems has been realized since Codd [16, 17] introduced the relational data model. Formally, a functional dependency (**fd**) is a statement, $X \rightarrow Y$ where X and Y are sets of attributes. A relation r satisfies this **fd** if for all t_1 and t_2 in r, $t_1[X] = t_2[X]$ implies $t_1[Y] = t_2[Y]$.

The design theory of relational databases is also concerned with the problem of finding a set of relation schemes that have lossless join. In order to define lossless join, let $\rho = \{R_1, R_2, \ldots, R_k\}$ be a set of relation schemes with $R(A_1A_2 \ldots A_n) = R_1R_2 \ldots R_k$. The project-join mapping defined by ρ, written as $\mathbf{m}_\rho$, is a function of relations r over R defined by

$$\mathbf{m}_\rho(r) = \mathbf{P}_{R_1}(r) \bowtie \mathbf{P}_{R_2}(r) \bowtie \cdots \bowtie \mathbf{P}_{R_k}(r) \tag{2.3}$$

The lossless join condition with respect to a set of dependencies D can be expressed as:

$$\text{for all } r \text{ satisfying } D,\ r = \mathbf{m}_\rho(r) \tag{2.4}$$

Aho et al. [1] have proposed an algorithm to test lossless join decomposition in the presence of functional and multivalued dependencies. For this purpose a tableau **T** is constructed [1, 29, 30, 46], which is a set of rows, best pictured as a matrix with one column for each attribute in the set of attributes R and row **i** corresponds to the relation scheme R_i. The element of **T** on row **i** and column **j** is a distinguished variable $\mathbf{a}_j$ if A_j is in R_i, otherwise the symbol $\mathbf{b}_{ij}$ is used. A set of transformation rules has been defined for different types of data dependencies that are used for changing a tableau **T** to a tableau **T***. Such rules are essentially a means to incorporate information about the set of admissible instances into the tableau.

In the algorithm proposed by Aho et al. [1] (to be referred to as the ABU algorithm), initially the tableau $\mathbf{T}_0$ associated with the given decomposition is constructed. The transformation rules corresponding to the data dependencies $d \in D$ are then repeatedly applied to $\mathbf{T}_0$ to generate a sequence of tableaus $\mathbf{T}_0$, $\mathbf{T}_1, \ldots, \mathbf{T}_s$ such that $\mathbf{T}_s = \mathbf{T}_{s+1}$ and no transformation rule can be applied further. This process is called *chase process* and $\mathbf{T}_s$ is denoted by $\mathbf{CHASE}_D(\mathbf{T}_0)$. The join is lossless if $\mathbf{CHASE}_D(\mathbf{T}_0)$ contains a row with all distinguished variables. The chase process has been applied to infer data dependencies. In fact, testing lossless join decomposition by this algorithm is equivalent to inferring the join dependency $\bowtie (R_1, R_2, \ldots, R_k)$ from D [29, 30, 46].

3. FUZZY RELATIONAL DATA MODEL

In this section we first introduce a few definitions and concepts from the fuzzy set theory literature. We then examine how fuzzy relations can be used to represent imprecise data. For a detailed discussion of fuzzy set theory and fuzzy calculus the reader is referred to [20, 26, 50–56].

3.1 Fuzzy Set

Let U be a classical set of objects, called the universe of discourse. An element of U is denoted by u.

Definition 3.1. A fuzzy set F in a universe of discourse U is characterized by a membership function

$$\mu_F\colon U \rightarrow [0, 1] \tag{3.1}$$

where $\mu_F(u)$ for each $u \in U$ denotes the grade of membership of u in the fuzzy set F.

Following the notations used in fuzzy set theory, we write

$$F = \{\mu(u_1)/u_1, \mu(u_2)/u_2, \ldots, \mu(u_n)/u_n\} \tag{3.2}$$

where $u_i \in U$, $1 \leq i \leq n$. Note that a classical subset A of U can be viewed as a fuzzy subset with membership function μ_A taking binary values, i.e.,

$$\begin{aligned} \mu_A(u) &= 1 \quad \text{if} \quad u \in A \\ &= 0 \quad \text{if} \quad u \notin A \end{aligned}$$

The usual set theory operations (such as union, intersection and complimentation, etc.) have been extended to deal with fuzzy sets. Let A and B be two fuzzy subsets in a universe of discourse with membership functions μ_A and μ_B, respectively. Then membership functions of $A \cup B$, $A \cap B$ and $\bar{A}$ (complement of A) are as given below.

$$\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u)) \tag{3.3}$$

$$\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u)) \tag{3.4}$$

$$\mu_{\bar{A}}(u) = 1 - \mu_A(u) \tag{3.5}$$

Based on these definitions, most of the properties that hold for classical set operations, such as DeMorgan's Laws, have been shown to hold for fuzzy sets. The only law of ordinary set theory that is no longer true is the law of the excluded middle, i.e.,

$$A \cap \bar{A} \neq \varnothing \quad \text{and} \quad A \cup \bar{A} \neq U \tag{3.6}$$

where $\varnothing$ is the null set, i.e., $\mu_\varnothing(u) = 0$ for all $u \in U$.

Given two fuzzy subsets A and B in U, B is a fuzzy subset of A, denoted by $B \subseteq A$, if

$$\mu_B(u) \leq \mu_A(u) \qquad \text{for all} \quad u \in U. \tag{3.7}$$

Two fuzzy sets A and B are said to be equal if $A \supseteq B$ and $A \subseteq B$.

In order to define Cartesian product of fuzzy sets, let $U = U_1 \times U_2 \times \cdots \times U_n$ be the Cartesian product of n universes and $A_1, A_2, \ldots, A_n$ be fuzzy sets in $U_1, U_2, \ldots, U_n$, respectively. The Cartesian product $A_1 \times A_2 \times \cdots A_n$ is defined to be a fuzzy subset of $U_1 \times U_2 \times \cdots \times U_n$ where

$$\mu_{A_1 \times \cdots \times A_n}(u_1 \ldots u_n) = \min(\mu_{A_1}(u_1), \ldots, \mu_{A_n}(u_n)) \tag{3.8}$$

where $u_i \in U_i$, $i = 1, \ldots, n$. Finally, given a fuzzy set, it is often necessary to construct a classical set with elements having membership value greater than $\alpha \in [0, 1]$. Thus, given a fuzzy set A in U, the α-cut of A is given by

$$A_\alpha = \{u \mid u \in U \text{ and } \mu_A(u) \geq \alpha\} \tag{3.9}$$

The definition of fuzzy set in (3.1) has also been extended to define a category of fuzzy sets [20]. For instance, type-2 fuzzy sets are defined to be fuzzy sets whose grades of membership are themselves fuzzy, i.e., μ_F in (3.1) is a mapping from U to the sets of fuzzy sets over [0, 1].

3.2 Possibility Distribution and Fuzzy Sets

Instead of treating $\mu_F(u)$ to be the grade of membership of u in F, one may interpret it as a measure of the possibility that a variable X has a value u, where X takes values in U. As for example, consider the fuzzy set HIGH-SALARY given below.

$$\begin{aligned}\text{HIGH-SALARY} = \{&0.1/20{,}000,\ 0.3/30{,}000,\ 0.5/40{,}000,\\ &0.7/50{,}000,\ 0.9/70{,}000,\ 0.95/80{,}000,\ 1.0/90{,}000\}\end{aligned} \tag{3.10}$$

Suppose it is known that John has a "high salary." Then, according to the possibilistic interpretation, one concludes that the possibility of John having salary = 30,000, is 0.3. Zadeh [53, 54] has suggested that a fuzzy proposition **X is F**, where F is a fuzzy subset of U and X is a variable which takes values from U, induces a possibility distribution Π_X that is equal to F, i.e.,

$$\Pi_X = F \tag{3.11}$$

The possibility assignment equation (3.11) is interpreted as

$$\text{Poss}(X = u) = \mu_F(u) \quad \text{for all } u \in U \tag{3.12}$$

Thus the possibility distribution of X is a fuzzy set, which serves to define the possibility that X could have any specified value u in U. One may also define a function $\pi_X\colon U \to [0, 1]$ that is equal to μ_F and associates with each $u \in U$ the possibility that X could take u as its value, i.e.,

$$\pi_X(u) = \text{Poss}(X = u) \quad \text{for } u \in U \tag{3.13}$$

The function π_X is called the possibility distribution function of X. The possibility distribution Π_X may also be used to define a fuzzy measure Π on U, where for any $A \subseteq U$,

$$\Pi(A) = \text{Poss}(X \in A) = \sup_{u \in A} \Pi_X(u) \tag{3.14}$$

For further details on possibility distribution and on the difference between possibility and probability measures, the reader is referred to [20, 53, 54]. In Section 4, we discuss the rules of fuzzy calculus that can be used to compute the possibility distribution of a compound fuzzy proposition from the possibility distributions of its constituent atomic fuzzy propositions.

3.3 Fuzzy Relations

Mathematically, an n-ary fuzzy relation r is a fuzzy subset of the Cartesian product of some universes. Thus, given $U_1, U_2, \ldots, U_n$ of n universes, a fuzzy relation r is a fuzzy subset of $U_1 \times U_2 \times \cdots \times U_n$ and is characterized by the n-variate membership function [20, 26, 54]

$$\mu_r\colon U_1 \times U_2 \times \cdots \times U_n \to [0, 1] \tag{3.15}$$

While applying this definition to relational databases, it is necessary to provide appropriate interpretation for the elements of U_i, $i = 1, \ldots, n$ and μ_r. For this purpose, we note that in a relational data model that can support imprecise information, it is necessary to accommodate two types of impreciseness—namely, the impreciseness in data values and impreciseness in the association among data values. As an example of impreciseness in data values, consider the Employee(Name, Salary) database, where Salary of an employee, say John, may be known to the extent that it lies in the range \$60,000–80,000, or may be known as John has a "high salary." Similarly, as an example of impreciseness in the association among data values, let Likes(Student, Course) represent how much a student likes a particular course. Here the data values may be precisely known, but the degree to which a student, say John, likes DBMS is imprecise. It is also not difficult to envisage examples where both ambiguity in data values as well as impreciseness in the association among them are present.

In recent years, there have been some attempts to use fuzzy set theory and related concepts for providing a suitable interpretation of different types of impreciseness in relational databases. Buckles and Petry [9–13] have suggested that attribute values be replaced by sets of values. A fuzzy similarity measure has also been used to identify similar tuples. Ruspini [42] has used a lattice organization for domains, where domain values correspond to one or more lattice points determined by a possibility distribution. Umano [47, 48] and Prade and Testemale [34] have proposed models based explicitly on possibility distributions where domain values are taken from sets of possibility distributions and associations among entities are also measured by possibility distributions. Prade [33] and Prade and Testemale [34] have shown that such an extended data model can accommodate different types of "null values" used in classical relational database literature. Baldwin [3] has used a mixed approach, where domain values are allowed to be fuzzy sets and association among entities is represented as a truth value in [0, 1]. Zemankova-Leech and Kandel [57] have provided a good exposition of the fuzzy relational data model and have advocated the use of linguistic quantifiers.

In the present treatment of the fuzzy relational data model, we will try to adhere to the notations used in classical relational database theory as far as possible. Thus a relation scheme R is a finite set of attribute names $\{A_1, A_2, \ldots, A_n\}$ and will be denoted by $R(A_1A_2 \ldots A_n)$ or simply by R. Corresponding to each attribute name A_i, $1 \leq i \leq n$, is a set $\text{dom}(A_i)$, called the domain of A_i. However, unlike classical relations, in the fuzzy relational model, $\text{dom}(A_i)$ may be a fuzzy set or even a set of fuzzy sets. Hence, along with each attribute A_i, we associate a set U_i, called the universe of discourse for the domain values of A_i.

Definition 3.2. A fuzzy relation r on a relation scheme $R(A_1A_2 \ldots A_n)$ is a fuzzy subset of $\text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$.

Depending on the complexity of $\text{dom}(A_i)$, $i = 1, \ldots, n$, we classify fuzzy relations into two categories. In type-1 fuzzy relations, $\text{dom}(A_i)$ can only be a fuzzy set (or a classical set). A type-1 fuzzy relation may be considered as a first-level extension of classical relations, where we will be able to capture the impreciseness in the association among entities. The type-2 fuzzy relations provide further generalization by allowing $dom(A_i)$ to be even a set of fuzzy sets (or possibility distributions). By enlarging $dom(A_i)$, type-2 relations enable us to represent a wider type of impreciseness in data values. Such relations can be considered as a second-level generalization of classical relations.

Finally, like classical relations, a fuzzy relation r will be represented as a table with an additional column for $\mu_r(t)$ denoting the membership value of the tuple t in r. Moreover, this table will contain only those tuples for which $\mu_r(t) > 0$, i.e., for any tuple not present in the table we assume $\mu_r(t) = 0$. This observation can be considered as a fuzzy version of the closed world assumption [38]. Since the law of excluded middle (3.6) does not hold for fuzzy sets, if $\mu_{\bar{r}}(t) > 0$, where $\bar{r}$ is the complement of r, we cannot conclude $\mu_r(t) = 0$. Only when $\mu_{\bar{r}}(t) = 1$, i.e., t is definitely in $\bar{r}$, do we have $\mu_r(t) = 0$.

Type-1 *Fuzzy Relational Data Model.* As mentioned above, in type-1 fuzzy relations, $dom(A_i)$ may be a classical subset or a fuzzy subset of U_i. Let the membership function of $dom(A_i)$ be denoted by μ_{A_i}, for $i = 1, \ldots, n$. Then from the definition of the Cartesian product of fuzzy sets in (3.8), $dom(A_1) \times dom(A_2) \times \cdots \times dom(A_n)$ is a fuzzy subset of $U = U_1 \times U_2 \times \cdots \times U_n$. Hence a type-1 fuzzy relation r is also a fuzzy subset of U with membership function μ_r. Also from (3.7) and (3.8), for all $(u_1 u_2 \ldots u_n) \in U$, μ_r must satisfy

$$\mu_r(u_1 u_2 \ldots u_n) \leq \min(\mu_{A_1}(u_1), \mu_{A_2}(u_2), \ldots, \mu_{A_n}(u_n)) \tag{3.16}$$

According to possibilistic interpretation of fuzzy sets, μ_r can be treated as a possibility distribution function in U. Thus $\mu_r(u_1 u_2 \ldots u_n)$ determines the possibility that a tuple $t \in U$ has $t[A_i] = u_i$, for $i = 1, \ldots, n$. In other words, $\mu_r(u_1 u_2 \ldots u_n)$ is a fuzzy measure of association among a set of domain values $\{u_1, u_2, \ldots, u_n\}$.

Example 3.1. Consider a relation scheme LIKES(Student, Course), where dom(Student) and dom(Course) are ordinary sets, i.e., domain values are crisp. In the fuzzy relation r shown in Table I, $\mu_r(t)$ can be interpreted as a possibility measure of a student liking a particular course. Thus the possibility of John liking DBMS is 0.90. So μ_r is a fuzzy measure of the association between Student and Course.

It is also possible to provide an alternative interpretation of μ_r as a fuzzy truth value belonging to [0, 1]. According to this interpretation, for a tuple t, $\mu_r(t)$ is the truth value of a fuzzy predicate associated with r when the variables in the predicate are replaced by $t[A_i]$, $i = 1, \ldots, n$.

Example 3.2. Consider a relation scheme R (N, H, X, S) of highly experienced and highly salaried employees, where N = Employee's name, J = Job, X = Experience, and S = Salary. Here, dom(N) and dom(J) are ordinary sets. But dom(X) and dom(S) are the fuzzy sets High-Experience and High-Salary in appropriate universes. Suppose that the universe of discourse U_X for the Experience is the set of positive integers in the range 0–30. Similarly, U_S, the universe of discourse for Salary, is the set of integer numbers in the range 10,000–100,000.

Table I. An Instance r of LIKES

Student	Course	μ
John	DBMS	0.90
Mary	DBMS	0.70
John	AI	0.80
Ashok	AI	0.95

Table II. An Instance r of Highly Experienced and Highly Salaried Employees

Name	Job	Experience	Salary	μ
John	Engineer	8	60,000	0.67
Ashok	Manager	9	70,000	0.80
Mary	Secretary	8	40,000	0.50
James	Engineer	12	80,000	1.00
Robin	Engineer	9	60,000	0.80

The membership function μ_{HX} and μ_{HS} of the fuzzy sets High-Experience and High-Salary are as given below.

$$\begin{aligned} \mu_{HX}(x) &= (1 + |x - 10|/4)^{-1} && \text{for} \quad x \le 10 \\ &= 1 && \text{for} \quad x > 10 \end{aligned} \qquad (3.17)$$

$$\begin{aligned} \mu_{HS}(s) &= (1 + |s - 60{,}000|/20{,}000)^{-1} && \text{for} \quad s \le 60{,}000 \\ &= 1 && \text{for} \quad s > 60{,}000 \end{aligned} \qquad (3.18)$$

Note that the membership function associated with the fuzzy set descriptor **high** is domain dependent. A typical instance r of R is shown in Table II.

In this example, $\mu_r(t)$ can be interpreted as the truth value of the fuzzy proposition "Y has high experience and high salary" for the tuple t. Thus the truth value of the fuzzy proposition "John has high experience and high salary" is 0.67.

In many applications, it may be necessary to combine both these interpretations of the membership function. For instance, in an entity relationship (E-R) model [19, 46], one may interpret μ_r as the possibility of association among the entities and follow truth value interpretation for membership of a tuple in the entity sets. In this connection, a recent paper by Zvieli and Chen [58] may be referred to, where fuzzy set theory has been applied to extend the E-R model and basic operations of fuzzy E-R algebra have been examined. An alternative interpretation of the membership function is also useful in supporting views of the data from several perspectives.

Type-2 fuzzy relational data model. Although type-1 relations enable us to represent impreciseness in the association among data values, its role in capturing uncertainty in data values is rather limited. For example, in a type-1 relational model for Employee(Name, Salary), one is not permitted to specify salary of John to be in the range \$40,000–\$50,000 and that of Mary to be a fuzzy set "low." With a view to accommodating a wider class of data ambiguities, we next consider a further generalization of the fuzzy relational data model where for any attribute

A_i, dom(A_i) may be a set of fuzzy sets in U_i. As a consequence of this generalization, a tuple $t = (a_1 a_2 \ldots a_n)$ in $\mathbf{D} = \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$ becomes a fuzzy subset of $U = U_1 \times U_2 \times \cdots \times U_n$ with

$$\mu_t(u_1 u_2 \ldots u_n) = \min[\mu_{a_1}(u_1), \mu_{a_2}(u_2), \ldots, \mu_{a_n}(u_n)] \tag{3.19}$$

where $u_i \in U_i$, for $i = 1, \ldots, n$. Since (3.19) holds for all $u_i \in U_i$, $i = 1, \ldots, n$, and according to Definition 3.2, a type-2 fuzzy relation r is a fuzzy subset of $\mathbf{D}$, from (3.7), the membership function

$$\mu_r: \mathbf{D} \to [0, 1] \tag{3.20}$$

must satisfy the following condition:

$$\mu_r(t) \le \max_{(u_1 u_2 \ldots u_n) \in U} [\min\{\mu_{a_1}(u_1), \mu_{a_2}(u_2), \ldots, \mu_{a_n}(u_n)\}] \tag{3.21}$$

where $t = (a_1 a_2 \ldots a_n) \in \mathbf{D}$.

As in the case of type-1 relations, μ_r may be interpreted either as a possibility measure of association among the data values or as a truth value of a fuzzy predicate associated with r. Regarding the interpretation of a fuzzy data value $a_i \in \text{dom}(A_i)$, we can treat a_i as a possibility distribution on U_i. In other words, for a tuple $t = (a_1 a_2 \ldots a_n) \in \mathbf{D}$, the possibility of $t[A_i] = u_i$ is equal to $\mu_{a_i}(u_i)$. For example, suppose that an instance of the relation Employee(Name, Salary) contains a tuple (John, S), where $S = \{0.3/10{,}000, 0.6/20{,}000, 0.8/30{,}000\}$. Here S represents the possibility distribution for the salary of John, i.e., Poss(Salary of John = 20,000) = 0.6.

Based on the possibilistic interpretation, for a tuple t of r, we obtain

$$\begin{aligned} \text{Poss}(t[A_1] = u_1, t[A_2] = u_2, \ldots, t[A_n] = u_n) \\ = \min\{\mu_r(t), \mu_t(u_1 u_2 \ldots u_n)\} \end{aligned} \tag{3.22}$$

where $u_i \in U_i$, $i = 1, \ldots, n$ and μ_t is given by (3.19). It is also possible to extend (3.22) to find the possibility that for a tuple $t = (a_1 a_2 \ldots a_n)$, $t[A_i] = b_i$, where b_i is a fuzzy subset of U_i. Evaluation of such a condition is, however, related to the concept of compatibility of two fuzzy propositions [20, 53–56].

Example 3.3. Let us consider the relation EMPLOYEE(N, D, J, X, S, I), where N = Name of the Employee, D = Department, J = Job, X = Experience, S = Salary, and I = Income tax. The dom(N), dom(D), and dom(J) are ordinary sets. But dom(X), dom(S), and dom(I) are sets of fuzzy sets in universes U_X, U_S, and U_I, respectively. As in Example 3.2, U_X, U_S, and U_I are assumed to be sets of positive integers in the range 0–30, 10,000–100,000, and 0–10,000, respectively. A typical instance r of Employee is shown in Table III, where fuzzy set descriptors High, Low, Moderate, etc., have been used to represent fuzzy data values over respective domains. Since all the elements of a classical subset have a membership value of 1.0, for notational convenience, elements of classical subsets are represented without their membership values, such as {10, 14, 19} instead of {1.0/10, 1.0/14, 1.0/19}. Also, a classical set of consecutive integers such as {10, 11, 12} is denoted by 10–12. The membership functions of the fuzzy set descriptors High, Low, etc., are domain dependent and are as given below.

Table III. An Instance r of EMPLOYEE Relation

Name	Department	Job	Experience	Salary	Income tax	μ
Murty	Mechanical	Engineer	10	50,000	5,000	0.70
Roy	Electrical	Manager	15–20	High	High	0.90
Kumar	Accounts	Accountant	Little	Low	Low	0.60
John	Sales	Manager	Moderate	40,000–60,000	4,000–7,000	0.80

For $x \in U_X$,

$$\begin{aligned} \mu_{\text{Moderate}}(x) &= (1 + |x - 8|)^{-1} && \text{for } x > 1 \\ &= 0 && \text{otherwise,} \end{aligned}$$

and

$$\begin{aligned} \mu_{\text{Little}}(x) &= (1 + 12x)^{-1} && \text{for } x > 0 \\ &= 0 && \text{otherwise.} \end{aligned}$$

Similarly,

$$\begin{aligned} \mu_{\text{High}}(y) &= (1 + a|y - c|)^{-1} && \text{for } y \le c \\ &= 1 && \text{for } y > c, \end{aligned}$$

where, $a = 1/20{,}000$, $c = 60{,}000$ for $y \in U_S$, and $a = 1/1000$, $c = 5000$ for $y \in U_I$.

Also, $\mu_{\text{Low}}(y) = 1 - \mu_{\text{High}}(y)$

Applying (3.22) to the second tuple in r, we can conclude that the possibility of Roy having Experience = 18, Salary = 50,000, and Income tax = 5,000 is 0.67. The possibility value thus obtained would be useful during query evaluation for identifying the tuples that have nonzero (or greater than a given threshold) possibility of satisfying the query predicate.

The type-2 relational model described here has some similarities with the fuzzy relational models considered by Buckles and Petry [12], Baldwin [3], Prade and Testemale [34], and Umano [47, 48]. The heterogeneous data model proposed in [12] allows domain values to be fuzzy sets, but no membership value is attached to a tuple. The fuzzy relational model suggested in [34, 47, 48] primarily uses a possibilistic interpretation. Even the membership value $\mu_r(t)$ of a tuple is treated as a possibility distribution in [0, 1] (see [48]). The implementation of fuzzy relational databases—especially the design of a query language that can support fuzzy constructs and the evaluation of such queries—have been examined in [3, 11, 12, 14, 15, 25, 34, 48, 49, 54, 57, 58]. In a subsequent paper [37], we will discuss the implementation of a fuzzy relational database system using the concept of abstract data type. Incidentally, the idea of using abstract data types to enhance the semantic knowledge in a relational database system has been suggested by Stonebraker [44].

3.4 Fuzzy Relational Operations

The relational algebra introduced by Codd [16, 17] consists of traditional set operations such as union, intersection, cross product, etc., and some special relational operations such as projection, join, etc. Since a fuzzy relation is, by definition, a fuzzy subset of the Cartesian product of its attribute domains, the definitions of union, intersection, and cross-product of fuzzy sets discussed in Section 3.1 can also be applied to fuzzy relations [9, 20, 34, 47, 48, 53, 57].

Table IV. Projection of an Instance of Highly Experienced and Highly Salaried Employees

Job	Salary	μ
Engineer	60,000	0.8
Engineer	80,000	1.0
Manager	70,000	0.8
Secretary	40,000	0.5

Extension of specialized relational algebra operations to deal with fuzzy relations have been considered by Zadeh and others [12, 34, 47, 48, 53, 54, 57]. Here, we discuss projection and join of fuzzy relations as these operations are of primary importance in the study of the lossless join problem.

Let the fuzzy relation r be an instance of a relation scheme $R(A_1A_2 \ldots A_n)$. Consider a subset $R_i(A_{i_1} \ldots A_{i_k})$ of R. As in Section 2, for a tuple t of r (i.e., $\mu_r(t) > 0$), $t[R_i]$ denotes the restriction of t on the attributes of R_i. Thus for $t = (a_1a_2 \ldots a_n)$, $t[R_i] = (a_{i_1} \ldots a_{i_k})$.

According to Zadeh [53, 54], the projection $r_i = \mathbf{P}_{R_i}(r)$ is a k-ary fuzzy relation in $\text{dom}(A_{i_1}) \times \cdots \times \text{dom}(A_{i_k})$. Also the membership function μ_{r_i} is given by

$$\mu_{r_i}(t) = \max_{\mathbf{t}_r} \{\mu_r(\mathbf{t}_r) \mid \mathbf{t}_r[R_i] = t\} \tag{3.23}$$

where $\mathbf{t}_r$ is a tuple of r and $t \in \text{dom}(A_{i_1}) \times \cdots \times \text{dom}(A_{i_k})$. Thus the tuples of r_i are the restrictions of the tuples of r, as in the case of classical relations. The max operator in (3.23) ensures that if more than one tuple in r, say $S_t \subseteq r$, has the same restriction t on R_i, then the projection r_i contains only one tuple and its membership value is the maximum of the grades of the tuples in S_t. In the case of classical relations, since grades have binary values, (3.23) will only lead to duplicate removal. Umano [47, 48] has suggested an extension of (3.23) when $\mu_r(t)$ is treated as a possibility distribution in [0, 1].

Example 3.4. The projection of the fuzzy relation r in Table II over R_{JS} = {Job, Salary} is shown in Table IV.

In the fuzzy set literature, projection is also called marginal fuzzy restriction [20, 53]. As a converse operation, Zadeh [53, 54] has defined the *cylindrical extension* of a fuzzy relation.

Let the fuzzy relation r_i be an instance of a relation scheme $R_i(A_{i_1} \ldots A_{i_k})$. Also consider a relation scheme $R(A_1A_2 \ldots A_n)$ where $R_i \subseteq R$. The cylindrical extension of r_i on R is denoted by $\mathbf{C}_R(r_i)$. According to Zadeh [53, 54], the cylindrical extension $\hat{r}_i = \mathbf{C}_R(r_i)$, is an n-ary fuzzy relation in $\mathbf{D} = \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$. The membership function $\mu_{\hat{r}_i}$ of $\hat{r}_i$ is given by

$$\mu_{\hat{r}_i}(t) = \mu_{r_i}(t[R_i]) \quad \text{for } t \in \mathbf{D} \tag{3.24}$$

From the definitions of projection and cylindrical extension, it follows that for any instance r of a relation scheme R and $R_i \subseteq R$,

$$\mu_{\hat{r}_i}(t) \geq \mu_r(t) \tag{3.25}$$

Table V. *An Instance r_1 of TEACHING*

Teacher	Course	μ
Rao	DBMS	0.80
Rao	AI	0.60
Johnson	DBMS	0.60
Johnson	AI	0.90

where $\hat{r}_i = \mathbf{C}_R(\mathbf{P}_{R_1}(r))$ and $t \in \mathbf{D}$. In other words,

$$r \subseteq \mathbf{C}_R(\mathbf{P}_{R_i}(r)) \tag{3.26}$$

Referring to Examples 3.1 and 3.4, for the tuple t = (John Engineer 8 60,000), $\mu_r(t) = 0.67$, whereas in the cylindrical extension $\hat{\mathbf{r}}_{JS}$ of $P_{R_{JS}}$, $\mu_{\hat{r}_{JS}}(t) = 0.80$. In fact, we may arrive at situations where $\mu_r(t) = 0$, i.e., the tuple definitely does not belong to r, yet $\mu_{\hat{r}_i}(t) \neq 0$, for some $\hat{r}_i = \mathbf{C}_R(\mathbf{P}_{R_i}(r))$. For instance, according to Example 3.1, μ_r(John Engineer 10 80,000) = 0. But this tuple definitely belongs to $\hat{\mathbf{r}}_{JS}$, i.e., $\mu_{r_{JS}}$(John Engineer 10 80,000) = 1.0. As will be seen later, this observation will play an important role in the lossless join of fuzzy relations.

It can also be established that for any instance r_i of R_i and $R \supseteq R_i$,

$$r_i = \mathbf{P}_{R_i}(\mathbf{C}_R(r_i)) \tag{3.27}$$

We are now in a position to define the **join**(natural) of fuzzy relations. Let $\rho = \{R_1, R_2, \ldots, R_s\}$ be a set of relation schemes and $R(A_1A_2 \ldots A_n) = R_1R_2 \ldots R_s$. Consider a set of fuzzy relations $\{r_1, r_2, \ldots, r_s\}$, where r_i is an instance of R_i, $i = 1, \ldots, s$. The natural join of these fuzzy relations written as

$$r = r_1 \bowtie r_2 \bowtie \cdots \bowtie r_s \tag{3.28}$$

(or simply as $\bowtie_{i=1}^{s} r_i$) is a fuzzy relation of the relation scheme R.

The membership function of r is given by [34, 47, 53, 54].

$$\begin{aligned}\mu_r(a_1a_2 \ldots a_n) = \min[&\mu_{\hat{r}_1}(a_1a_2 \ldots a_n),\\ &\mu_{\hat{r}_2}(a_1a_2 \ldots a_n), \ldots \mu_r(a_1a_2 \ldots a_n)]\end{aligned} \tag{3.29}$$

where $a_i \in \mathrm{dom}(A_i)$ and $\hat{r}_j$ is the cylindrical extension of r_j on R, for $i = 1, \ldots, n$ and $j = 1, \ldots, s$.

Example 3.5. Let us consider the join of the instance r of the relation LIKES in Table I with an instance r_1 of TEACHING(Teacher, Course) shown in Table V. The membership function μ_{r_1} is interpreted as a measure of the teacher's ability to teach a course.

The relation **r** obtained by taking the natural join of these two relations is shown in Table VI. The grade values $\mu_r(t)$ of tuples in **r** can be interpreted as a measure of association among students and teachers based on student's liking for a course and teacher's ability to teach that course.

When the fuzzy relations r_i, $i = 1, \ldots, s$ are obtained by taking the projections of a fuzzy relation r on R_i, $i = 1, \ldots, s$, from (3.26) and (3.29) we have

$$r \subseteq \mathbf{P}_{R_1}(r) \bowtie \mathbf{P}_{R_2}(r) \bowtie \cdots \bowtie \mathbf{P}_{R_s}(r) \tag{3.30}$$

Table VI. Natural Join of Fuzzy Relations in Table I and Table V

Student	Teacher	Course	μ
John	Rao	DBMS	0.80
John	Johnson	DBMS	0.60
John	Rao	AI	0.60
John	Johnson	AI	0.80
Mary	Rao	DBMS	0.70
Mary	Johnson	DBMS	0.60
Ashok	Rao	AI	0.60
Ashok	Johnson	AI	0.90

that is,

$$r \subseteq \mathbf{m}_\rho(r) \tag{3.31}$$

where the project join mapping $\mathbf{m}_\rho(r) = \mathbf{P}_{R_1}(r) \bowtie \mathbf{P}_{R_2}(r) \bowtie \cdots \bowtie \mathbf{P}_{R_n}(r)$.

The condition (3.30) holds for classical relations [29, 46] and leads to the lossless join problem.

4. FUZZY INTEGRITY CONSTRAINTS

The integrity constraints in relational database systems can be broadly classified into two groups [19, 29, 46]:

(1) *Domain dependency*—which restricts admissible domain values of the attributes, e.g., "age of an employee is less than 65 years," or "no one is 10 feet tall."

(2) *Data dependency*—which requires that if some tuples in the database fulfill certain equalities, then either some other tuples must also exist in the database, or some values of the given tuples must be equal.

Among these two types of dependencies, data dependencies have received wider attention as they have greater impact on the design of the database systems. Several types of data dependencies, such as functional dependency, multivalued dependency, join dependency, etc., have been identified and the associated implication problem has been examined [1, 4–6, 19, 21, 22, 24, 29, 30, 43, 46]. The implication problem of data dependencies is the problem of deciding whether a given set of dependencies logically implies another dependency and has important bearing on the automated synthesis of database schemes.

As we generalize relational database systems to deal with fuzzy or incomplete information, it will be necessary to consider integrity constraints that involve fuzzy constructs. Thus in a relation PLAYERS(Name, Age, Height, Sport, Income), an integrity constraint may be stated as, "Most basketball players are tall," or "Many tennis players have high income." These integrity constraints impose restrictions on the admissible values of height or income of the basketball or tennis players, respectively. Similarly, as an example of a fuzzy data dependency, consider the relation scheme EMPLOYEE(Name, Department, Job, Experience, Salary), where an integrity constraint may be stated as "in any department employees having similar jobs and experience must have almost equal salary."

Table VII. An Instance of Young and Intelligent Students

Name	Age	Height	Course	Marks	μ
Ashok	25	175	DBMS	75	0.75
Ashok	25	175	AI	90	0.80
John	23	170	DBMS	92	0.90
John	23	170	AI	85	0.85
Sheila	20	160	DBMS	70	0.70

Zadeh [53, 54] has introduced the concept of *particularization* to deal with data constraints. The particularization of a fuzzy relation r of a relation scheme $R(A_1A_2 \ldots A_n)$ is the effect of specification of the possibility distribution of one or more attributes $Y \subseteq R$. The resulting relation $\mathbf{r}$ is therefore a restriction of the original relation. Even the *select* operator in relational algebra can be treated as a special case of particularization of relations.

According to Zadeh [53, 54], particularization may be viewed as the result of forming the conjunction of a fuzzy proposition **X is F** with the particularizing proposition **Y is G** where $X = A_1A_2 \ldots A_m$, $Y = A_{i_1} \ldots A_{i_k} \subseteq X$, F is a fuzzy subset of $\text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_m)$, and G is a fuzzy subset of $\text{dom}(A_{i_1}) \times \cdots \times \text{dom}(A_{i_k})$. The conjunction of these two fuzzy propositions is expressed in terms of the possibility distributions induced by them. As in Section 3.2, we associate possibility distributions $\Pi_X = F$ and $\Pi_Y = G$ with the fuzzy propositions **X is F** and **Y is G**, respectively. The particularization of Π_X by Π_Y (or equivalently F by G) is denoted by $\Pi_X[\Pi_Y = G]$ and is defined to be the intersection of F and G, i.e.,

$$\Pi_X[\Pi_Y = G] = F \cap \hat{G} \tag{4.1}$$

where $\hat{G}$ is the cylindrical extension of G in $\text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_m)$.

Thus the particularization of a fuzzy relation r of $R(A_1A_2 \ldots A_n)$ due to a fuzzy proposition **Y is G**, where $Y = A_{i_1} \ldots A_{i_k}$, produces the relation $\mathbf{r}$ with membership function as given below.

$$\mu_{\mathbf{r}}(a_1a_2 \ldots a_n) = \min[\mu_r(a_1a_2 \ldots a_n), \mu_{\hat{G}}(a_1a_2 \ldots a_n)] \tag{4.2}$$

where $a_i \in \text{dom}(A_i)$ and $\hat{G}$ is the cylindrical extension of G. The fuzzy proposition **Y is G** may either represent an integrity constraint, as in Example 4.1 given below, or may be the predicate of a select operation on the database.

Example 4.1. In Table VII, an instance of young and intelligent students with relation scheme R(Name, Age, Height, Course, Marks) is shown.
Now suppose that this relation must satisfy the fuzzy domain dependency "Students are Tall," where the fuzzy set tall is as given below.

$$\text{Tall} = \{0.40/150, 0.50/160, 0.65/165, 0.80/170, 0.90/175, 1.00/180\}$$

The particularization of the fuzzy relation in Table VII due to this fuzzy proposition produces an instance of the fuzzy relation young, intelligent, and tall students shown in Table VIII, where the membership values are computed using Table VII.

Table VIII. Particularization of the Fuzzy Relation in Table VII

Name	Age	Height	Course	Marks	μ
Ashok	25	175	DBMS	75	0.75
Ashok	25	175	AI	90	0.80
John	23	170	DBMS	92	0.80
John	23	170	AI	85	0.80
Sheila	20	160	DBMS	70	0.50

4.1 Translation Rules of Fuzzy Calculus

In order to evaluate the particularization of a fuzzy relation due to a compound fuzzy proposition, it is necessary to examine how the possibility distribution of a compound fuzzy proposition can be obtained from the possibility distributions of its constituent atomic propositions. For this purpose, the following rules, called translation rules, of fuzzy calculus developed by Zadeh [53–55] may be used.

Suppose F and G are fuzzy subsets of the universes U and V, respectively. As in Section 3.2, with atomic propositions **X is F** and **Y is G** we associate possibility distributions Π_X and Π_Y, respectively, where $\Pi_X = F$, and $\Pi_Y = G$.

T1. *Modifier Rule.* Consider the modified proposition **X is σF**, where σ is a modifier, such as "not," "very," or "more or less." Each modifier is related to a function f_σ: $[0, 1] \rightarrow [0, 1]$. The possibility distribution Π_X^* of the modified proposition **X is σF** is given by

$$\Pi_X^* = F^* \tag{4.3}$$

Here, F^* is a fuzzy subset of U with membership function

$$\mu_{F^*}(u) = f_\sigma(\mu_F(u)) \qquad \text{for} \quad u \in U \tag{4.4}$$

Thus the effect of the modifier is to generate a new possibility distribution on U that is uniquely determined by the modification function f_σ associated with the modifier and the possibility distribution function μ_F of the atomic proposition. Following modification functions are recommended in the fuzzy set literature for some commonly used modifiers [20, 53–55].

$$\sigma = \text{not},\ f_\sigma(x) = 1 - x \tag{4.5}$$

$$\sigma = \text{very},\ f_\sigma(x) = x^2 \tag{4.6}$$

$$\sigma = \text{more or less},\ f_\sigma(x) = \sqrt{x} \tag{4.7}$$

T2. *Composition Rules.* These rules can be used to find the possibility distribution associated with a compound proposition of the type **X is F $\odot$ Y is G**, where the composition operator $\odot$ may correspond to "and," "or," etc. The possibility distribution $\Pi(X \odot Y)$ associated with such a compound proposition is given by

$$\Pi(X \odot Y) = F \odot G \tag{4.8}$$

where $F \odot G$ is a fuzzy subset of $U \times V$. Depending on the composition operator, the fuzzy subset $F \odot G$ is as given below.

$$\odot = \text{and}, \qquad F \odot G = \hat{F} \cap \hat{G} \tag{4.9}$$

$$\odot = \text{or}, \qquad F \odot G = \hat{F} \cup \hat{G} \tag{4.10}$$

where $\hat{F}$ and $\hat{G}$ are cylindrical extensions of F and G in $U \times V$, respectively, and union and intersection of fuzzy sets have been defined in Section 3.1.

For a conditional fuzzy proposition **If X is F then Y is G**, Zadeh [53] has used a translation rule called the compositional rule of inference. This rule is based on the definition of implication in Lukasiewicz's multivalued logic [40]. However, Fukami et al. [23] have pointed out that the consequences inferred by the compositional rule of inference often do not fit our intuition and do not even satisfy quite natural criteria such as modus ponens, modus tollens, or syllogism. In fuzzy logic, alternative translation rules for conditional fuzzy propositions have been proposed [20, 31]. These translation rules are usually based on different implication rules of multivalued logic [40]. Mizumoto and Zimmerman [32] have compared the translation rules for conditional fuzzy proposition and have shown that the translation rule based on the implication in *standard sequence logic* (called R_s), or the *Godelian implication rule* (called R_g) satisfy modus ponens, modus tollens, and syllogism. But even these translation rules may fail to satisfy some generalizations of modus ponens or modus tollens.

In this paper, we will use the translation rule R_s defined below to determine the possibility distribution associated with a conditional fuzzy proposition.

Definition 4.1. Let F and G be fuzzy subsets of U and V, respectively. The possibility distribution $\Pi(X \to Y)$ associated with the conditional fuzzy proposition **If X is F then Y is G** is given by

$$\Pi(X \to Y) = R_s \tag{4.11}$$

where R_s is a fuzzy subset of $U \times V$ with membership function

$$\begin{aligned} \mu_{R_s}(u, v) &= 1 \qquad \text{if} \quad \mu_F(u) \le \mu_G(v) \\ &= 0 \qquad \text{otherwise} \end{aligned} \tag{4.12}$$

Note μ_{R_s} defines a hard partition of $U \times V$, i.e., R_s is an ordinary subset of $U \times V$.

Besides the translation rules discussed above, Zadeh [53–55] has also proposed translation rules for quantified fuzzy propositions, such as **QX is F**, where the quantifier Q may represent "most," "many," "some," "few," etc. Similarly, the translation rules to determine the possibility distribution of a qualified fuzzy proposition **X is F is** τ have also been suggested. The qualifier τ may involve truth qualification, such as "true" or "very true," or a possibility/probability qualification.

In a relational database system, if the integrity constraints involve compound fuzzy propositions, the translation rules discussed above can be applied to determine the possibility distribution induced by such integrity constraints. We may then apply (4.2) to accommodate the effects of these restrictions on any fuzzy relation.

5. FUZZY FUNCTIONAL DEPENDENCY AND INFERENCE RULES

Our next objective is to extend the design theory of relational database systems to the fuzzy domain. In this quest, it is necessary to study fuzzy data dependencies and their associated implication problem. In this paper, we concentrate on the extension of functional dependencies as they constitute the most important class among the different data dependencies that have been identified so far and it is easier to identify such dependencies from design specifications.

As mentioned in Section 2, an instance r of a relation scheme $R(A_1 A_2 \ldots A_n)$ satisfies a functional dependency f: $X \rightarrow Y$ if, for each pair of tuples t_1 and t_2 of r such that $t_1[X] = t_2[X]$, we have $t_1[Y] = t_2[Y]$. In the fuzzy domain, equality of domain values defines a fuzzy proposition and may even be specified as "approximately equal," "more or less equal," etc. For instance, a fuzzy data dependency in the relation EMPLOYEE(Name, Job, Experience, Salary) can be stated as "Job and Experience more or less determines Salary."

5.1 Equality as a Fuzzy Relation

The fuzzy relation EQUAL(EQ) defined below can be used as a fuzzy measure to compare elements of a given domain.

Definition 5.1 A fuzzy relation EQUAL (EQ) over a universe of discourse U is defined to be a fuzzy subset of $U \times U$, where μ_{EQ} satisfies the following conditions. For all $a, b \in U$,

$$\begin{aligned} &\mu_{EQ}(a, a) = 1 && \text{(reflexivity)} \\ &\mu_{EQ}(a, b) = \mu_{EQ}(b, a) && \text{(symmetry)} \end{aligned} \tag{5.1}$$

That is, EQUAL is a fuzzy resemblance relation over U [20, 26]. In terms of possibility theory, $\mu_{EQ}(a, b)$ can be interpreted as the possibility of treating a and b as "equal." The membership function μ_{EQ} should be appropriately selected during database creation to capture the meaning of equality/approximate equality of domain values as perceived by the database designer.

Remark. It may be noted that unlike the classical equality, EQUAL is not assumed to be transitive, i.e., EQUAL need not be a similarity relation [20, 26]. This has been done with the objective of capturing integrity constraints arising out of approximate equality of domain values. In fact, with most distance/proximity measures [20, 26] used for compairing domain values, transitivity does not hold.

We can extend the definition of EQUAL over composite domains as follows. Let $\mathbf{D} = \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$, and t_1, t_2 be two tuples in $\mathbf{D}$. Suppose that in each $\text{dom}(A_i)$, a fuzzy resemblance relation EQUAL with membership function μ^i_{EQ} has already been defined to compare the elements of the domain. Then the fuzzy relation EQUAL, extended over $\mathbf{D}$, defines a fuzzy subset of $\mathbf{D} \times \mathbf{D}$, with the membership function as given below

$$\begin{aligned} \mu_{EQ}(t_1, t_2) = \min\{&\mu^1_{EQ}(t_1[A_1], t_2[A_1]), \\ &\mu^2_{EQ}(t_1[A_2], t_2[A_2]), \ldots, \mu^n_{EQ}(t_1[A_n], t_2[A_n])\} \end{aligned} \tag{5.2}$$

In type-1 fuzzy relations, $dom(A_i)$ may be a fuzzy subset of the universe U_i. We can define resemblance of the elements of $dom(A_i)$ by

$$\mu^{i}_{EQ}(a, b) = \min\{\mu_{\alpha}(a, b), \mu_{\theta}(\mu_{A_i}(a), \mu_{A_i}(b))\} \tag{5.3}$$

where $a, b \in U_i$, μ_{α} and μ_{θ} are the membership functions of resemblance relations over U_i and $[0, 1]$, respectively. It can be readily seen that μ_{EQ} defined by (5.3) is reflexive and symmetric. Similarly, in a type-2 relation, we may define EQUAL over $dom(A_i)$ by

$$\mu^{i}_{EQ}(a_{i1}, a_{i2}) = \min_{u_i \in U_i} \psi(\mu_{a_{i1}}(u_i), \mu_{a_{i2}}(u_i)) \tag{5.4}$$

where $a_{i1}, a_{i2} \in dom(A_i)$ are fuzzy subsets of U_i and ψ is a resemblance relation over [0, 1]. In the fuzzy set literature, several resemblance relations have been examined [20, 26]. For example, $\theta(x_1, x_2)$ may be selected as $1 - | x_1 - x_2 |$, or as $1/(1 + | x_1 - x_2 |)$. It should also be mentioned that the definitions (5.3) and (5.4) are not unique ways of defining EQUAL over the domains of type-1 or type-2 relations (for instance, see Example 5.1 given below).

Example 5.1. Consider the relation scheme EMPLOYEE(N, D, J, X, S, I) and its instance r discussed in Example 3.3. In this case, some of the domain values are crisp, whereas dom(X), dom(S), and dom(I) are sets of fuzzy subsets. We select the following membership functions as measures of equality over different domains.

(1) $\mu_{EQ}(a, b) = 0$ for $a \neq b$, $a, b \in dom(A)$, $A \in \{N, D, J\}$. In other words, names of the employees, departments, or jobs must exactly match to qualify for equality.

(2) For $a, b \in dom(A)$, $A \in \{X, S, I\}$, the following situations may arise.

  (i) The domain values a, b are both crisp (i.e., single element fuzzy set with binary membership value). In that case,

$$\begin{aligned} \mu_{EQ}(a, b) &= (1/(1 + \beta\, | a - b |), \\ \text{where } \beta &= 1 \text{ for } a, b \in dom(X), \\ &= 1/2000 \text{ for } a, b \in dom(S), \\ &= 1/500 \text{ for } a, b \in dom(I). \end{aligned}$$

  (ii) If a is crisp and b is a fuzzy subset, than $\mu_{EQ}(a, b) = \mu_b(a)$. Similarly, if b is crisp and a is a fuzzy subset, then $\mu_{EQ}(a, b) = \mu_a(b)$. (Note, an ordinary subset such as 15–20 can be considered as a fuzzy subset with binary membership value.)

  (iii) If both a and b are fuzzy subsets, then

  $\mu_{EQ}(a, b) = \max\{\mathbf{c}/card(a), \mathbf{c}/card(b)\}$,

  where *card* denotes cardinality of a fuzzy set [20, 26, 55], and $\mathbf{c} = card(a \cap b)$.

It can be readily checked that the membership functions defined above are reflexive and symmetric, as demanded by (5.1). Also, over dom(A), $A \in \{X, S, I\}$, the fuzzy resemblance relation EQUAL can be used to provide interpretation of approximate equality of the domain values. In Example 5.2, we will see how these fuzzy resemblance relations can be used to represent a fuzzy data dependency "for any job, experience determines salary."

Once the meaning of equality of domain values is finalized, the modifier rule of fuzzy calculus can be applied to provide interpretation for fuzzy relations "more or less equal" or "very much equal." For instance, the membership function of the fuzzy relation "more or less equal" would be given by

$$\mu_{\mathrm{MLEQ}}(a, b) = \sqrt{\mu_{\mathrm{EQ}}(a, b)} \tag{5.5}$$

In general, the membership function associated with a modified relation σEQUAL in a dom(A) is given by

$$\mu_{\sigma\mathrm{EQ}}(a, b) = f_\sigma(\mu_{\mathrm{EQ}}(a, b)) \tag{5.6}$$

where f_σ is the modifier function associated with σ and $a, b \in$ dom(A). In order that σEQUAL be reflexive, we require $f_\sigma(1) = 1$. The symmetry of σEQUAL follows from that of EQUAL. This modified membership function can now be applied to determine the possibility of, say, two salaries to be more or less equal.

Finally, it may be mentioned that although the idea of treating equality as a fuzzy relation is motivated by our attempt to generalize integrity constraints, a similar approach can be followed in query processing. For example, a database query may be specified as "Get the name of employees who have more or less equal salary as their manager." In this case, "equal" should be treated as a fuzzy resemblance relation, and "more or less" as a modifier of "equal." Also, the fuzzy relation "equal" during query processing may be described by a different membership function than the one used for compairing domain values in fuzzy data dependencies. The evaluation of this query will retrieve the names of employees who have nonzero (or above a certain threshold value) possibility of having salary "more or less equal" to the salary of their manager. The possibility value of the tuple to be selected will be determined by the membership function of the fuzzy relation "more or less equal." In a subsequent paper [37], we will deal with the extension of QUEL[45] and associated query evaluation procedures to support this type of query on a fuzzy relational database.

5.2 Fuzzy Functional Dependency

Let $X = A_{i_1}A_{i_2} \ldots A_{i_k}$ and $Y = A_{j_1}A_{j_2} \ldots A_{j_p}$ be subsets of a relation scheme $R(A_1A_2 \ldots A_n)$. Based on our interpretation of "equality," a fuzzy proposition **X is equal** defines a fuzzy subset of $\mathrm{dom}(A_{i_1}) \times \mathrm{dom}(A_{i_2}) \times \ldots \mathrm{dom}(A_{i_k})$ with the membership function determined by (5.2). A generalization of an **fd:** $X \rightarrow Y$ in R, called **ffd:** $X \rightsquigarrow Y$, can now be defined as a particularization of a fuzzy relation on R due to a fuzzy conditional proposition **If X is equal then Y is equal**. The particularization imposed by an **ffd** is, therefore, linked with the translation rule used for conditional fuzzy propositions. In the following development we use the translation rule given in Definition 4.1.

With a fuzzy proposition **If X is equal then Y is equal**, the possibility distribution determined by (4.12) produces a hard partition of $\mathrm{dom}(A_1) \times \mathrm{dom}(A_2) \times \cdots \times \mathrm{dom}(A_n)$. Combining this observation with the concept of particularization, we arrive at the following definition of a fuzzy functional dependency.

Definition 5.2. A fuzzy functional dependency (**ffd**) X $\rightsquigarrow$ Y with X, Y $\subseteq$ R, holds in a fuzzy relation r on R, if for all tuples t_1 and t_2 of r (i.e., $\mu_r(t_i) > 0$, for i = 1, 2), we have

$$\mu_{EQ}(t_1[X], t_2[X]) \leq \mu_{EQ}(t_1[Y], t_2[Y]) \tag{5.7}$$

Remark. An **fd** in a classical relational database can be viewed as a special case of an **ffd**. To prove this, consider the **ffd**: X $\rightsquigarrow$ Y and suppose that the resemblance relation EQUAL over dom(A), A $\in$ XY, satisfies the additional property that $\mu_{EQ}(a, b) = 0$ for $a \neq b$, a, $b \in$ dom(A). Since for a classical relational database r, $\mu_r(t) = 1$ for t $\in$ r, Definition 5.2 implies that no two tuples of r can agree in X-values, yet disagree in their Y values.

Example 5.2. Let us again consider the relation scheme EMPLOYEE(N, D, J, X, S, I) and the resemblance relations discussed in Example 5.1. It can be verified that with these choices for μ_{EQ}, the fuzzy relation r in Table III satisfies the following fuzzy functional dependencies.

ND $\rightsquigarrow$ J: Name and Department determines Job.

JX $\rightsquigarrow$ S: For any job, employees having "equal" experience should have "equal" salary.

S $\rightsquigarrow$ I: For "equal" salaries, income taxes are "equal."

Here the **ffd**: ND $\rightsquigarrow$ J is in fact, a classical **fd** due to the special nature of the EQUAL relation over the domains of these attributes. Note that the **ffd**: JX $\rightsquigarrow$ S does not permit the tuple t = (Wilson Electrical Engineer 11 55,000 5700) to be inserted in the database because r already contains the tuple t_1 = (Murty Mechanical Engineer 10 50,000 5,000) and

$$\mu_{EQ}(t[JX], t_1[JX]) > \mu_{EQ}(t[S], t_1[S])$$

However, insertion of this tuple would not violate a classical **fd**: JX $\longrightarrow$ S. Thus the integrity constraint "for any job, experience determines salary" is not adequately represented by a classical **fd** when interpretation of this integrity constraint requires that "for any job, employees having approximately equal experience must have approximately equal salary." By suitably selecting μ_{EQ}, the **ffd**: JX $\rightsquigarrow$ S provides a more acceptable model for such integrity constraints.

The proposed approach to generalization of functional dependencies also enables us to capture integrity constraints involving suitable fuzzy modifiers to "equal." Given a fuzzy modifier σ with $f_\sigma(1) = 1$, we have seen that the modified relation σEQUAL is also reflexive and symmetric. With such modifiers, an integrity constraint defined by a fuzzy proposition **If X is σ_1 equal then Y is σ_2 equal** can be regarded as an **ffd**, where the modified membership functions of EQUAL should be used in (5.7), i.e., as

$$f_{\sigma_1}(\mu_{EQ}(t_1[X], t_2[X])) \leq f_{\sigma_2}(\mu_{EQ}(t_1[Y], t_2[Y])) \tag{5.8}$$

Thus with the integrity constraint "X more or less determines Y," we associate a modified equality relation MLEQ over the domains of the attributes in Y,

Table IX. An Instance of SUPPLY

Item #	Item	Order-Date	Delivery-Date
100	nut	15. 10. 86	18. 10. 86
102	bolt	12. 10. 86	18. 10. 86
100	nut	20. 10. 86	24. 10. 86
104	nail	15. 10. 86	18. 10. 86

where the membership function of MLEQ is computed using (5.5). Observe that the modified relation MLEQ is also reflexive and symmetric as required in Definition 5.1. It can now be said that the **ffd**: "X more or less determines Y" holds in a fuzzy relation r, if for any two tuples t_1, t_2 of r,

$$\mu_{EQ}(t_1[X], t_2[X]) \leq \sqrt{\mu_{EQ}(t_1[Y], t_2[Y])} \tag{5.9}$$

Example 5.3. Suppose that a relation SUPPLY(Item#, Item, Order-Date, Delivery-Date) satisfies an integrity constraint "for any item, Order-Date more or less determines Delivery-Date." In order to represent this integrity constraint we consider the following resemblance relations over the domains of Item, Order-Date, and Delivery-Date.

(i) $\mu_{EQ}(a, b) = 0$ for $a \neq b$, a, $b \in$ dom(Item).
(ii) $\mu_{EQ}(a, b) = 1/(1 + |a - b|)$ for a, $b \in$ dom(Order-Date), or dom(Delivery-Date),

where $|a - b|$ represents the difference in number of days between two dates a and b.

Then, from (5.5), "more or less equal delivery date" defines a fuzzy resemblance relation over dom(Delivery-Date) with membership function $\sqrt{1/(1 + |a - b|)}$. The **ffd** representing the given integrity constraint requires that in any instance of SUPPLY, any two tuples t_1 and t_2 with $t_1[\text{Item}] = t_2[\text{Item}]$ must satisfy

$$\begin{aligned} &1/(1 + |t_1[\text{Order-Date}] - t_2[\text{Order-Date}]|) \\ &\quad \leq \sqrt{(1/(1 + |t_1[\text{Delivery-Date}] - t_2[\text{Delivery-Date}]|)} \end{aligned} \tag{5.10}$$

The classical relation shown in Table IX is a typical instance of SUPPLY that satisfies (5.8). This **ffd** does not permit a tuple (102 bolt 14. 10. 86 29. 10. 86) to be inserted in the database because, along with the tuple (102 bolt 12. 10. 86 18. 10. 86), already present in the database, such insertion will violate (5.10). The insertion of this tuple, however, does not violate a classical **fd**: Item Order-Date $\rightarrow$ Delivery-Date thereby implying that the **fd** has failed to capture the meaning of the given integrity constraint.

Finally, we would like to mention that the Definition 5.2 of a fuzzy functional dependency is not a unique way of generalizing classical **fd** in a fuzzy database. For instance, with an alternative translation rule for conditional fuzzy propositions, such as the one based on the *Godelian implication rule* [20, 32, 40], a different set of conditions is obtained. Secondly, Definition 5.2 requires that the validity of an **ffd** in a fuzzy relation have binary truth values. Thus a relation may either satisfy the **ffd** X $\rightsquigarrow$ Y, or will contain tuples for which the condition (5.7) is violated. However, within the framework of fuzzy calculus it is possible to treat truth value of a fuzzy proposition to be a fuzzy subset of [0, 1] [20, 53, 54]. Hence, one can define a generalization of **fds** where the possibility of a fuzzy relation satisfying the given integrity constraint is even truth qualified. Some alternative generalizations of functional dependencies in a fuzzy database may be found in [13, 34]. Different formulations of fuzzy functional dependencies will be examined in a subsequent paper.

5.3 Inference Axioms for Fuzzy Functional Dependency

Given a set of data dependencies that hold on a database, it is often possible to derive other data dependencies that also hold on the same database [5, 6, 19, 29, 46]. In order to derive such new data dependencies from the given set of dependencies, a set of inference axioms is generally used. In the relational database literature, sound and complete sets of inference axioms for different types of data dependencies have been reported [19, 29, 46]. We next present a set of sound and complete inference axioms for **ffds**, which is similar to Armstrong's Axioms for classical **fds** [29, 46].

Let us consider a relation scheme $R(A_1, A_2, \ldots, A_n)$ and a set of **ffds** F. An instance r of R is called a legal instance if r satisfies all **ffds** in F. As mentioned in the previous section, the validity of an **ffd** in a fuzzy relation r will be assumed to have binary truth value. In the following axioms, X, Y, and Z are subsets of the relation scheme R.

Inference Axioms:

FF1. *Reflexivity*: If $Y \subseteq X$, then $X \rightsquigarrow Y$.

FF2. *Augmentation*: If $X \rightsquigarrow Y$ holds, then $XZ \rightsquigarrow YZ$ also holds.

FF3. *Transitivity*: If $X \rightsquigarrow Y$ and $Y \rightsquigarrow Z$ hold, then $X \rightsquigarrow Z$ holds.

The following inference axioms follow from above axioms.

FF4. *Union*: If $X \rightsquigarrow Y$ and $X \rightsquigarrow Z$ hold, then $X \rightsquigarrow YZ$ holds.

FF5. *Decomposition*: If $X \rightsquigarrow YZ$ holds, then $X \rightsquigarrow Y$ and $X \rightsquigarrow Z$ hold.

FF6. *Pseudotransitivity*: If $X \rightsquigarrow Y$ and $YW \rightsquigarrow Z$ hold, then $XW \rightsquigarrow Z$ holds.

To prove the soundness of these axioms, consider an instance r of R and let t_1 and t_2 be two tuples of r.

FF1. *Reflexivity*: Since Y is a subset of X, from (5.2) we have,

$$\mu_{EQ}(t_1[X], t_2[X]) \leq \mu_{EQ}(t_1[Y], t_2[Y])$$

So the **ffd**: $X \rightsquigarrow Y$ trivially holds in r.

FF2. *Augmentation*: Since $X \rightsquigarrow Y$ holds in the fuzzy relation r,

$$\mu_{EQ}(t_1[X], t_2[X]) \leq \mu_{EQ}(t_1[Y], t_2[Y])$$

Hence,

$$\begin{aligned}\min\{\mu_{EQ}(t_1[X], t_2[X]), \mu_{EQ}(t_1[Z], t_2[Z])\}\\ \leq \min\{\mu_{EQ}(t_1[Y], t_2[Y]), \mu_{EQ}(t_1[Z], t_2[Z])\}\end{aligned}$$

that is, $XZ \rightsquigarrow YZ$ holds in r.

FF3. *Transitivity*: Suppose r satisfies the **ffds** X $\rightsquigarrow$ Y and Y $\rightsquigarrow$ Z, that is,

$$\mu_{EQ}(t_1[X], t_2[X]) \leq \mu_{EQ}(t_1[Y], t_2[Y])$$

and

$$\mu_{EQ}(t_1[Y], t_2[Y]) \leq \mu_{EQ}(t_1[Z], t_2[Z])$$

These two conditions imply that

$$\mu_{EQ}(t_1[X], t_2[X]) \leq \mu_{EQ}(t_1[Z], t_2[Z])$$

Thus X $\rightsquigarrow$ Z also holds in r.

The remaining axioms follow from these three axioms, as in the case of classical **fds**.

The following definition, regarding the closure of a set of attributes with respect to a set of **ffds**, is similar to the corresponding definition used in the classical **fd** literature [29, 46].

Definition 5.3. Suppose F is a set of **ffds** of a relation scheme R and let W $\subseteq$ R. Then W^+, the closure of W with respect to F, is the set of attributes A $\in$ R, such that W $\rightsquigarrow$ A can be obtained from F using **ffd** inference axioms (FF1–FF6).

LEMMA 5.1. W $\rightsquigarrow$ V *follows from the inference axioms of* **ffds**, *iff* V $\subseteq W^+$.

PROOF. Let V = $\{A_1, \ldots, A_k\}$ and suppose that W $\rightsquigarrow$ V follows from the **ffd** inference axioms. Then for each i, W $\rightsquigarrow$ A_i holds by the Decomposition axiom. So, V $\subseteq W^+$.

Suppose, V $\subseteq W^+$. By Definition 5.3, for each i, W $\rightsquigarrow$ A_i is implied by the **ffd** axioms. Hence, according to the Union rule, W $\rightsquigarrow$ V follows. □

The following lemma will be useful in the study of the lossless join decomposition of fuzzy relations.

LEMMA 5.2. *Natural join of fuzzy relations preserves fuzzy functional dependencies.*

PROOF. Let r_1 and r_2 be two fuzzy relations and F_1 and F_2 be two sets of **ffds** satisfied by r_1 and r_2, respectively. Let $r = r_1 \bowtie r_2$. We show that r satisfies all the **ffds** in $F_1 \cup F_2$.

Consider the **ffd**, f: X $\rightsquigarrow$ Y in $F_1 \cup F_2$ and suppose that r does not satisfy f. Then r contains two tuples t_1 and t_2 such that

$$\mu_{EQ}(t_1[X], t_2[X]) > \mu_{EQ}(t_1[Y], t_2[Y]) \tag{5.11}$$

Suppose that t_k^i is the projection of t_k over the attributes of r_i, where i, k = 1, 2. Since $\mu_r(t_k) > 0$, from (3.23) and (3.29), $\mu_{r_i}(t_k^i) > 0$ for i, k = 1, 2.

Also, $t_k^i[XY] = t_k[XY]$, if X and Y are attributes of r_i. When $f \in F_j$ (j $\in$ {1, 2}), X and Y are attributes of r_j. Then (5.11) leads to the contradiction that r_j violates $f \in F_j$, for j $\in$ {1, 2}. Hence r satisfies $F_1 \cup F_2$. □

It can be similarly shown that if r_i satisfies F_i, for i = 1, ..., s, then $r = \bowtie_{i=1}^{s} r_i$ satisfies $\bigcup_{i=1}^{s} (F_i)^+$. Here, F^+ denotes the closure of a set of **ffds** F

with respect to the inference axioms, i.e., the set of all **ffds** that can be derived from F using the inference axioms.

As a consequence of the similarity of the inference axioms for **fds** and **ffds**, the following lemma can be readily proved. This lemma suggests that we can apply the membership algorithms for **fds** (such as the one proposed by Beeri and Bernstein [4]) to determine if an **ffd** f can be inferred from a given set of **ffds** using FF1–FF6.

LEMMA 5.3. *Let F be a set of* **ffds** *and* $\hat{F} = \{V \rightarrow W \mid V \rightsquigarrow W \in F\}$. *Suppose by applying Armstrong's axioms to* $\hat{F}$, *the* **fd** $\hat{f}$: $X \rightarrow Y$ *can be inferred. Then the* **ffd** f: $X \rightsquigarrow Y$ *can be inferred from F by* **ffd** *axioms* FF1–FF6.

PROOF. The lemma is proved by induction on the number of steps in the inference of $\hat{f}$ from $\hat{F}$.

Basis. If $\hat{f} \in \hat{F}$, then by definition of $\hat{F}$, $f \in F$. Also Beeri and Bernstein [4] have shown that in any derivation of $\hat{f}$ from $\hat{F}$, the reflexivity axiom need not be used more than once. In fact, if $Y \subseteq X$, by reflexivity $X \rightarrow Y$ trivially follows from $\hat{F}$. But in this case, the reflexivity axiom of the **ffd** is also applicable and $X \rightsquigarrow Y$ would follow trivially from F.

Induction. Suppose the claim is true for a derivation of $\hat{f}$ with k steps. That is if $\hat{F}$ logically implies $\hat{f}$ in k steps, then f can be inferred from F by **ffd** axioms. Consider an $\hat{f}$: $X \rightarrow Y$, which can be inferred from $\hat{F}$ using $(k + 1)$ applications of Armstrong's axioms. Following Beeri and Bernstein [4], the reflexivity axiom will not be used since otherwise $X \rightarrow Y$ would have a derivation of length one.

Suppose in the $(k + 1)$th step of the derivation of $\hat{f}$, the augmentation axiom is used with $\hat{f}_1$: $V \rightarrow Z$. Hence, $\hat{F}$ derives $\hat{f}_1$ in k steps. Also, the derivation of $\hat{f}$ from $\hat{f}_1$ using augmentation rule requires that $X = VW$ and $Y = ZW$. By our assumption, the **ffd** f_1: $V \rightsquigarrow Z$ can be derived from F using **ffd** axioms. Now, augmenting both sides of f by W, the **ffd** augmentation axiom produces $X \rightsquigarrow Y$. Next, suppose that the transitivity axiom is used in the $(k + 1)$th step of derivation of $\hat{f}$. Then, there exists $\hat{f}_1$: $X \rightarrow V$ and $\hat{f}_2$: $V \rightarrow Y$, such that both $\hat{f}_1$ and $\hat{f}_2$ can be derived from $\hat{F}$ in less than or equal to k steps and $\hat{f}$ follows from these two **fds** by transitivity. By our assumption, the **ffds** f_1: $X \rightsquigarrow V$ and f_2: $V \rightsquigarrow Y$ can be derived from F using **ffd** axioms. Also, the transitivity axiom can be applied to these two **ffds** to obtain $X \rightsquigarrow Y$. □

We next examine the completeness of the **ffd** axioms. A set of inference axioms is said to be complete for a family of constraints, if for each set F from the family, the constraints that are implied by F are exactly those that can be derived from it using these inference axioms [46]. The following example shows that unlike Armstrong's axioms, the **ffd** inference axioms FF1–FF6 are not always complete.

Example 5.4. Let us select resemblance relations over $\mathrm{dom}(A_i)$, $i = 1, 2, 3$, such that in addition to (5.1), the following condition holds:

For all $a_{1i}, a_{1j} \in \mathrm{dom}(A_1)$ and for all $a_{rk}, a_{rp} \in \mathrm{dom}(A_r)$

$$r = 2, 3, \text{ if } a_{rk} \neq a_{rp}, \text{ then } \mu_{EQ}(a_{1i}, a_{1j}) > \mu_{EQ}(a_{rk}, a_{rp}) \tag{5.12}$$

With this choice of EQUAL, consider the **ffd** $A_1A_2 \rightsquigarrow A_3$ over $R(A_1A_2A_3)$. Let r be a legal instance of R. Since r satisfies $A_1A_2 \rightsquigarrow A_3$, for any two tuples t_1 and t_2 of r

$$\mu_{EQ}(t_1[A_1A_2], t_2[A_1A_2]) \le \mu_{EQ}(t_1[A_3], t_2[A_3]) \tag{5.13}$$

where $\mu_{EQ}(t_1[A_1A_2], t_2[A_1A_2])$ is evaluated using (5.2).

We now show that r also satisfies $A_2 \rightsquigarrow A_3$. To prove this, we note that when the proposition $\Psi \equiv ((t_1[A_1] \neq t_2[A_1])$ and $(t_1[A_2] = t_2[A_2]))$ is true, from (5.1) and (5.2), $\mu_{EQ}(t_1[A_1A_2], t_2[A_1A_2]) = \mu_{EQ}(t_1[A_1], t_2[A_1])$. By (5.12) and (5.13), we then have $t_1[A_3] = t_2[A_3]$. Hence by reflexivity of resemblance relations, r satisfies $A_2 \rightsquigarrow A_3$.

Similarly, when Ψ is false (i.e., either $t_1[A_1] = t_2[A_1]$, or $t_1[A_2] \neq t_2[A_2]$), from (5.2) and (5.12), $\mu_{EQ}(t_1[A_1A_2], t_2[A_1A_2]) = \mu_{EQ}(t_1[A_2], t_2[A_2])$. So from (5.13) and (5.7), r satisfies $A_2 \rightsquigarrow A_3$.

Therefore, for the present choice of EQUAL, the **ffd** $A_1A_2 \rightsquigarrow A_3$ implies $A_2 \rightsquigarrow A_3$ even though such an implication cannot be made using FF1–FF6. If we restrict EQUAL further by the condition that $\mu_{EQ}(a_{1i}, a_{1j}) = 1.0$ for all a_{1i}, $a_{1j} \in dom(A_1)$ (i.e., EQUAL does not distinguish the elements of $dom(A_1)$), r will trivially satisfy the **ffds** $A_2 \rightsquigarrow A_1$ and $A_3 \rightsquigarrow A_1$. Again, none of these two **ffds** can be inferred from $A_1A_2 \rightsquigarrow A_3$ using FF1–FF6.

Example 5.4 indicates that, depending upon the type of the resemblance relations used for defining the **ffds**, it is possible to imply new **ffds** that cannot be inferred using FF1–FF6. To infer such **ffds**, we have to consider additional inference axioms that depend on the resemblance relations used for comparing the domain values. Since the number of resemblance relations that can be defined over any domain is infinite (not even countably infinite), a complete set of inference axioms can be obtained only for suitable class of **ffds** where additional restrictions are imposed on EQUAL. In this connection, it will be useful to find a class of **ffds** for which the inference axioms FF1–FF6 constitute a complete set. The following theorem establishes completeness of these **ffd** axioms when each domain has at least two elements that do not resemble each other.

THEOREM 5.1. *The inference axioms* FF1, FF2, *and* FF3 *form a complete set of inference axioms for fuzzy functional dependencies of a relation scheme* $R(A_1A_2 \ldots A_n)$ *when the following condition holds*:

For each $A_i \in R$, *there exists at least one pair of elements* $a_i, b_i \in dom(A_i)$ *such that* $\mu_{EQ}(a_i, b_i) = 0$. (5.14)

PROOF. Given a set F of **ffds** over R. Suppose that the **ffd** f: $W \rightsquigarrow V$, with W, $V \subseteq R$, cannot be inferred from F using the inference axioms.

Let r be a fuzzy relation with two tuples t and t_1 shown below. $t = (a_1, a_2, \ldots, a_n)$, and the tuple t_1 is defined by

$$\begin{aligned} t_1[A_i] &= a_i \quad && \text{if } A_i \in W^+ \\ &= b_i && \text{otherwise} \end{aligned}$$

where $\mu_{EQ}(a_j, b_j) = 0$, for $a_j, b_j \in dom(A_j)$, $j = 1, \ldots, n$. From our assumption, such a_j and b_j exist in each attribute domain.

First we show that r satisfies all the **ffds** in F. Let $X \rightsquigarrow Y$ be an **ffd** in F. If $X \not\subseteq W^+$, then for $A_j \in (X - W^+)$, $\mu_{EQ}(t[A_j], t_1[A_j]) = \mu_{EQ}(a_j, b_j) = 0$. So from (5.2), $\mu_{EQ}(t[X], t_1[X]) = 0$. Hence, by (5.7) r trivially satisfies $X \rightsquigarrow Y$. When $X \subseteq W^+$, by Lemma 5.1 $W \rightsquigarrow X$ and by transitivity $W \rightsquigarrow Y$. Applying Lemma 5.1 again, $Y \subseteq W^+$. Since $XY \subseteq W^+$, by construction $\mu_{EQ}(t[X], t_1[X]) = \mu_{EQ}(t[Y], t_1[Y]) = 1.0$. Therefore r satisfies the **ffd** $X \rightsquigarrow Y$.

Since f cannot be inferred from F using the inference axioms, we now show that r does not satisfy f. From the definition of r, $t[W] = t_1[W]$. In order that r satisfies f, we must have $\mu_{EQ}(t[V], t_1[V]) = 1$. But this requires $V \subseteq W^+$, thereby implying that f follows from F by the inference axioms. Since this conclusion violates our original assumption, the fuzzy relation r cannot satisfy the **ffd**: $W \rightsquigarrow V$. Hence the inference rules are complete. □

The condition (5.14) defines a class of **ffds** for which the inference axioms FF1–FF3 constitute a complete set. In view of the similarity of the inference axioms, we would be able to apply many of the algorithms reported in the functional dependency literature to this class of **ffds**. For example, Beeri and Bernstein's algorithm [4] can be applied to find a minimal cover of a set of such **ffds**. Since in most real-world applications it is not difficult to select suitable resemblance relations that satisfy (5.14), the **ffds** belonging to this class can still represent fuzzy integrity constraints involving approximate equality of domain values. For instance, in Example 5.3, we may define the resemblance relations over dom(Order-Date) and dom(Delivery-Date) such that

$$\begin{aligned} \mu_{EQ}(a, b) &= 1/(1 + |a - b|) && \text{for } |a - b| \leq 30 \\ &= 0 && \text{otherwise.} \end{aligned}$$

The resulting **ffd** still captures the semantics of the integrity constraint "for any item, Order-Date more or less determines Delivery-Date" for any two orders placed within 30 days.

Since (5.14) is only a sufficient condition for the **ffd** axioms FF1–FF6 to be complete, it may, however, be worthwhile to find a restriction on EQUAL that is both necessary and sufficient for the completeness of these axioms.

6. LOSSLESS JOIN OF FUZZY RELATIONS

To answer user's queries in a fuzzy relational database it would often be necessary to join two or more fuzzy relations. However, (3.30) suggests that the natural join may not recover the original fuzzy relation. The problem of lossless join of relation schemes is of central importance in the design theory of relational databases. In fact, the concept of adequate decomposition of relational databases requires that the synthesized relation scheme have the lossless join property [29, 46]. In this section, we examine the lossless join decomposition of fuzzy relations in the presence of fuzzy functional dependencies.

Definition 6.1. Let R be a scheme and $\rho = \{R_1, R_2, \ldots, R_s\}$ be a decomposition of R with $R = R_1 R_2 \ldots R_s$. This decomposition is a lossless join with respect to a set of **ffds** F, if for every fuzzy relation r of R that satisfies these **ffds**, the following condition holds.

$$r = \mathbf{m}_\rho(r) \tag{6.1}$$

Table X. An Instance r of R(T,A,S,E)

Name	Age	Subject	Experience	μ
Rao	30	DBMS	4	0.8
Rao	30	AI	2	0.6
Johnson	35	AI	5	0.9
Sen	28	OS	3	0.7

It was established in [36] that a decomposition ρ of R is lossless join for a given set of integrity constraints iff, for any $(a_1a_2 \ldots a_n) \in \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$, there exists at least one $\hat{r}_i = \mathbf{C}_R(\mathbf{P}_{R_i}(r))$, $i \in \{1, 2, \ldots, s\}$, such that

$$\mu_{\hat{r}_i}(a_1a_2 \ldots a_n) = \mu_r(a_1a_2 \ldots a_n) \tag{6.2}$$

Note that the validity of the condition is required for any tuple $t \in \mathbf{D} = \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$, and not merely for those tuples that are already present in r, i.e., $\mu_r(t) > 0$. The following example, with a classical relation (i.e., a fuzzy relation with binary membership values), illustrates that a tuple t may not be present in an instance r, yet in none of its projections $\mu_{\hat{r}_i}(t) \neq 0$, thereby leading to lossy join.

Example 6.1. Consider a relation scheme R(ABC) with **fds** A $\rightarrow$ B and C $\rightarrow$ B, and an instance r of R having two tuples $t_1 = (a_1bc_1)$ and $t_2 = (a_2bc_2)$. Let $\rho = \{AB, BC\}$ be a decomposition of R. It is well known that this decomposition does not have lossless join [29, 46]. This conclusion can also be verified from (6.2) by observing that for a tuple $t = (a_1bc_2)$, $\mu_r(t) = 0$, yet both $\mu_{\hat{r}_1}(t) = 1$ and $\mu_{\hat{r}_2}(t) = 1$, where $\hat{r}_i$, $i = 1, 2$, are the cylindrical extensions of the projections of r on AB and BC, respectively.

In the next example, a lossless decomposition of a fuzzy relation based on classical **fds** is shown.

Example 6.2. Let R(T, A, S, E) be a relation scheme of experienced teachers, where T is the name of the teacher, A is the age, S is the subject taught, and E is the experience in teaching a particular subject. Here, the membership values can be interpreted as the possibility of a teacher teaching a particular subject. Suppose that classical **fds** T $\rightarrow$ A and TS $\rightarrow$ E hold in R. A typical legal instance r of R is shown in Table X. Let $\rho = \{TA, TSE\}$ be a decomposition of R. The projections of r over ρ are shown in Table XI(a and b). It can be easily checked that $r = \mathbf{P}_{TA}(r) \bowtie \mathbf{P}_{TSE}(r)$. Also, (6.2) can be verified by showing that $\mu_{\hat{r}_2}(t) = \mu_r(t)$ for all tuples $t \in \text{dom}(T) \times \text{dom}(A) \times \text{dom}(S) \times \text{dom}(E)$, where $\hat{r}_2$ is the cylindrical extension of the projection of r over the attributes T, S, and E [35].

Although (6.1) is a necessary and sufficient condition for lossless join decomposition of fuzzy relations, it cannot be applied, in practice, to test lossless join of a given decomposition as exhaustive testing with all possible combinations of domain values is required. However, it can be used to prove that a certain algorithm, such as the ABU algorithm for classical relations, can be applied to test the lossless join of relation schemes. In fact, using this condition Raju and

Table XI. Projection of r Over p = {TA, TSE}

Name	Age	μ
Rao	30	0.8
Johnson	35	0.9
Sen	28	0.7

(a)

Name	Subject	Experience	μ
Rao	DBMS	4	0.8
Rao	AI	2	0.6
Johnson	AI	5	0.9
Sen	OS	3	0.7

(b)

Majumdar [35] have shown that the ABU algorithm can indeed be applied to test the lossless join decomposition of fuzzy relations in the presence of classical **fds**. In this paper we will use this condition to prove some theorems concerning the lossless join decomposition of fuzzy relations in the presence of **ffds**.

6.1 Fuzzy Functional Dependency and Lossless Join Decomposition

In classical relational database theory, it is well known that if an **fd**: $X \rightarrow Y$ holds in a relational database R(XYZ), then the decomposition ρ = {XY, YZ} of R is lossless. This important property of **fds** forms the basis of Codd's normalization procedures and can be shown to follow from the necessary and sufficient condition for lossless join of a two-component decomposition obtained by Rissanen [41]. It can also be proved by ABU algorithm using **fd** transformation rules [29, 46]. We now show that unless the fuzzy resemblance relation EQUAL is appropriately restricted, a similar property does not hold for **ffds**.

Example 6.3. Consider a relation scheme R(ABC) with an **ffd** $A \rightsquigarrow B$. Also suppose that the membership functions of the fuzzy resemblance relation EQUAL used for defining this **ffd** do not distinguish the elements of the domains of the attributes A and B, i.e., $\mu_{EQ}(a, a_1) = 1$ and $\mu_{EQ}(b, b_1) = 1$, for all a, $a_1 \in$ dom(A) and b, $b_1 \in$ dom(B). Note, μ_{EQ} thus defined obviously satisfies (5.1). An instance r of R that satisfies this **ffd** is shown in Table XII.

Let R_1(AB) and R_2(AC) be the decompositions of R. The projections r_1 and r_2 over R_1 and R_2 are shown in Table XIII(a and b).

Let $\hat{r} = r_1 \bowtie r_2$. Then it can be checked that $\mu_{\hat{r}}(abc) = \mu_{\hat{r}}(ab_1c_1) = 0.9$, whereas $\mu_r(abc) = 0.8$ and $\mu_r(ab_1c_1) = 0.6$. In fact, $\hat{r} \supset r$, implying thereby that the decomposition of R based on the **ffd**: $A \rightsquigarrow B$ is not lossless.

It can further be shown that whenever the resemblance relation EQUAL over dom(B) has $\mu_{EQ}(b, b_1) = 1$ for any $b \neq b_1$, one can find an instance r of R that satisfies the **ffd**: $A \rightsquigarrow B$, yet the join of the projections of r over R_1 and R_2 is not equal to r.

Even though the assumption that EQUAL is a resemblance relation is sufficient for generalization of **fds** and associated inference axioms, this example suggests that for lossless synthesis of fuzzy relations with **ffds**, EQUAL needs to be

Table XII. An Instance *r* of R(ABC)

A	B	C	μ
a	b	c	0.8
a	b_1	c	0.9
a	b	c_1	0.9
a	b_1	c_1	0.6

Table XIII. Projections of *r* Over R_1(AB) and R_2(AC)

A	B	μ
a	b	0.9
a	b_1	0.9

(a)

A	C	μ
a	c	0.9
a	c_1	0.9

(b)

restricted further. In view of this, in the subsequent development the fuzzy resemblance relation EQUAL is restricted such that besides (5.1) it also satisfies

$$\mu_{EQ}(a, b) < 1 \quad \text{for} \quad a \neq b,\ a, b \in U \tag{5.1a}$$

Thus for $X \subseteq R(A_1A_2 \ldots A_n)$, (5.1a) and (5.2) would imply

$$\mu_{EQ}(t_1[X], t_2[X]) < 1, \quad \text{if} \quad t_1[X] \neq t_2[X] \tag{5.2a}$$

where $t_1, t_2 \in \mathbf{D} = \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$.

Hereinafter, the class of **ffds**, where EQUAL is restricted by (5.1a) will be referred to as restricted fuzzy functional dependency (**rffd**).

Note that if a fuzzy relation *r* satisfies an **rffd**: A $\rightsquigarrow$ B, then *r* cannot have two tuples which agree on A but disagree on B, just as in the case of classical **fds**. However, unlike classical **fds**, this **rffd** does not permit *r* to have two tuples t_1 and t_2 with $t_1[A] \neq t_2[A]$, if

$$\mu_{EQ}(t_1[A], t_2[A]) > \mu_{EQ}(t_1[B], t_2[B])$$

Thus whenever an **rffd** f: X $\rightsquigarrow$ Y holds in a fuzzy relation *r*, the **fd** $\hat{f}$: X $\rightarrow$ Y also holds in the same relation, although the converse is not true. The restriction imposed by (5.1a), therefore, makes an **rffd** a stronger integrity constraint than a classical **fd**. However, even with this additional restriction on EQUAL, the **ffds** are still useful in modeling integrity constraints arising out of approximate equality of domain values. For instance, the resemblance relations used for modeling the integrity constraint, "for any Item, Order-Date more or less determines the Delivery-Date" in Example 5.3, satisfy (5.1a). But some of the resemblance relations in Examples (5.1) and (5.2) do not satisfy (5.1a). For the relation R(N, D, J, X, S, I), in Example 5.2, to have lossless join decomposition ρ = {NDJ, JXS, SI}, we should select new resemblance relations over dom(X), dom(S), and dom(I), that satisfy (5.1a).

The following theorem shows that under this additional restriction, the **rffd**: $X \rightsquigarrow Y$ will lead to lossless join decomposition of R(XYZ).

THEOREM 6.1. *Given a relation scheme* $R(A_1A_2 \ldots A_n)$ *with an* **rffd**: $X \rightsquigarrow Y$, *where* $X, Y \subseteq R$. *The relation scheme* R *has a lossless join decomposition into two components* $R_1(XY)$ *and* $R_2(XZ)$, *where* $Z = R - XY$.

PROOF. Let r be a legal instance of R and r_1 and r_2 be the projections of R over R_1 and R_2, respectively. We denote the cylindrical extension of $r_i (i = 1, 2)$ on R by $\hat{r}_i$, and the natural join of r_1 and r_2 by $\hat{r}$.

In order that the given decomposition of R has lossless join, the condition (6.2) requires that for any $t \in \mathbf{D} = \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$, there exists at least one r_i, $i \in \{1, 2\}$, such that $\mu_r(t) = \mu_{\hat{r}_i}(t)$.

With a tuple t of r, i.e., $\mu_r(t) > 0$, by (3.25) we have

$$\mu_{\hat{r}_2}(t) \geq \mu_r(t) \tag{6.3}$$

However, for $\mu_{\hat{r}_2}(t) > \mu_r(t)$, r must have at least another tuple t_1 with $\mu_r(t_1) > \mu_r(t)$, such that $t[XZ] = t_1[XZ]$ and $t[Y] \neq t_1[Y]$. Then by (5.2a), $\mu_{EQ}(t[Y], t_1[Y]) < 1$, whereas from (5.1) and (5.2), $\mu_{EQ}(t[X], t_1[X]) = 1$. Hence the condition (5.7) for the **rffd**: $X \rightsquigarrow Y$ is violated. Thus, for all tuples t of r, we have $\mu_r(t) = \mu_{\hat{r}_2}(t)$, i.e., $\mu_{\hat{r}}(t) = \mu_r(t)$.

We now show that for a tuple $t \in \mathbf{D}$, which is definitely not in r, i.e., $\mu_r(t) = 0$, either $\mu_{r_1}(t[XY]) = 0$, or $\mu_{r_2}(t[XZ]) = 0$, i.e., $\mu_{\hat{r}}(t) = 0$. We prove this result by contradiction. Suppose for a tuple $t \in \mathbf{D}$, $\mu_r(t) = 0$, but both $\mu_{r_1}(t[XY]) \neq 0$ and $\mu_{r_2}(t[XZ] \neq 0$.

For $\mu_{r_1}(t[XY]) \neq 0$, (3.23) requires that there must exist a tuple t_1 in r (i.e., $\mu_r(t_1) > 0$) with $t_1[XY] = t[XY]$ and $t_1[Z] \neq t[Z]$. Similarly, for $\mu_{r_2}(t[XZ]) \neq 0$, we need a tuple t_2 in r, such that $t_2[XZ] = t[XZ]$ and $t_2[Y] \neq t[Y]$. In other words, the relation r must contain at least two tuples t_1 and t_2 such that $t_1[X] = t_2[X]$, yet $t_1[Y] \neq t_2[Y]$. But by (5.2a), this would violate the **rffd**: $X \rightsquigarrow Y$. We have, therefore, established that for any $t \in \mathbf{D}$, $\mu_r(t) = \mu_{\hat{r}}(t)$, i.e., the join is lossless. □

With classical **fds**, Rissanen [41] has also established a converse of Theorem 6.1, according to which, if $R_1(XY)$ and $R_2(XZ)$ is a lossless join decomposition of R(XYZ) with a set of **fds** F, then either $X \rightarrow Y$, or $X \rightarrow Z$, can be inferred from F using Armstrong's axioms. However, the following example shows that a similar result does not always hold with **rffds**.

Example 6.4. Consider a relation scheme R(ABC) with $\text{dom}(A) = \{a_1, a_2\}$, $\text{dom}(B) = \{b_1, b_2\}$ and $\text{dom}(C) = \{c_1, c_2\}$. Define resemblance relations over these domains such that, in addition to (5.1),

$$\mu_{EQ}(a_1, a_2) = 0.8, \qquad \mu_{EQ}(b_1, b_2) = 0.6 \qquad \text{and} \quad \mu_{EQ}(c_1, c_2) = 0.7.$$

Note that (5.1a) is also satisfied. With this choice of EQUAL, let us consider the **rffd** $AB \rightsquigarrow C$ and a decomposition $R_1(AB)$ and $R_2(BC)$ of R. Applying (6.2) to any legal instance r of R, it can be shown that this decomposition has lossless join. But neither $B \rightsquigarrow A$, nor $B \rightsquigarrow C$, can be inferred from $AB \rightsquigarrow C$ using FF1–FF6.

A closer examination reveals that as in Example 5.4, if an instance r of R satisfies AB $\rightsquigarrow$ C, then r also satisfies the **rffd** B $\rightsquigarrow$ C. In other words, for the present choice of EQUAL the **ffd** axioms FF1–FF6 are not complete.

This example suggests that the desired converse of Theorem 6.1 can hold only for a class of **rffds** where additional restrictions are imposed on EQUAL. In fact, when the resemblance relations of **rffds** are restricted further by (5.14), then by constructing a relation with two tuples (as in Theorem 5.1), it can be shown that a two-component decomposition has lossless join iff a condition similar to Rissanen [41] holds.

We now examine whether the ABU algorithm can be applied to test lossless join decomposition of fuzzy relations with **rffds**.

Consider a relation scheme R with a set F of **rffds** and let $\rho = \{R_1, R_2, \ldots, R_s\}$ be a decomposition of R. Construct a set of **fds** $\hat{F} = \{X \rightarrow Y \mid X \rightsquigarrow Y \in F\}$. As discussed before, whenever a relation r satisfies the set of **rffds** F, it also satisfies the **fds** in $\hat{F}$. Thus, we may say, the set of **ffds** F logically implies $\hat{F}$.

We may now apply the ABU algorithm to the decomposition ρ and let $\mathbf{T}_\rho^* = \mathbf{CHASE}_{\hat{F}}(\mathbf{T}_\rho)$ be the chase of $\mathbf{T}_\rho$ under $\hat{F}$, where $\mathbf{T}_\rho$ is the tableau associated with ρ [1, 29, 46]. If ρ has a lossless join with respect to $\hat{F}$, then as discussed in Section 2, $\mathbf{T}_\rho^*$ has a row (say ith row) which contains all distinguished variables [1, 29, 46]. It was shown in [35] that if the ith row of $\mathbf{T}_\rho^*$ consists of only distinguished variables, then the set of attributes R_i is a superkey of $R(R_i \rightarrow R)$. Thus the **fd** $R_i \rightarrow R$ can be inferred from $\hat{F}$ using Armstrong's axioms. By Lemma 3, we can then conclude that the **rffd**: $R_i \rightsquigarrow R$ can be inferred from F using **ffd** axioms. This observation enables us to prove the following theorem.

THEOREM 6.2. *Let* $\rho = \{R_1, R_2, \ldots, R_s\}$ *be a decomposition of a relation scheme* $R(A_1A_2 \ldots A_n)$ *and* F *be a set of* **rffds** *over* R. *If* $\mathbf{CHASE}_{\hat{F}}(\mathbf{T}_\rho)$, *where* $\hat{F} = \{X \rightarrow Y \mid X \rightsquigarrow Y \in F\}$ *is the set of* **fds** *implied by F and* $\mathbf{T}_\rho$ *is the tableau associated with* ρ, *has a row that contains only distinguished variables, then* ρ *has a lossless join.*

PROOF. Suppose that ith row of $\mathbf{CHASE}_{\hat{F}}(\mathbf{T}_\rho)$ contains only distinguished variables. Then, as noted above, the **rffd** f: $R_i \rightsquigarrow R$ can be inferred from F using **ffd** axioms. Consequently, any legal instance r of R that satisfies F, will also satisfy f.

Let $r_j = \mathbf{P}_{R_j}(r)$. The cylindrical extension of r_j on R is denoted by $\hat{r}_j$. Now consider a tuple t in r. By (3.23) and (3.25), $\mu_{\hat{r}_i}(t) \neq \mu_r(t)$ if and only if there exists another tuple t_1 in r such that $\mu_r(t_1) > \mu_r(t)$ and $t[R_i] = t_1[R_i]$, $t[R - R_i] \neq t_1[R - R_i]$.

Since EQUAL relations used in defining **rffds** satisfy (5.1a), with t and t_1, $\mu_{EQ}(t[R - R_i], t_1[R - R_i]) < 1$, whereas $\mu_{EQ}(t[R_i], t_1[R_i]) = 1$. Hence, the presence of both t and t_1 in r would violate the **ffd** $R_i \rightsquigarrow R$. Thus, for all tuples t of r, $\mu_{\hat{r}_i}(t) = \mu_r(t)$.

For a tuple $t \in \mathbf{D} = \text{dom}(A_1) \times \text{dom}(A_2) \times \cdots \times \text{dom}(A_n)$, which is definitely not in r, i.e., $\mu_r(t) = 0$, we show that there exists a projection r_j, $j \in \{1, 2, \ldots, s\}$, of r such that $\mu_{\hat{r}_j}(t) = 0$.

To prove this result by contradiction, suppose $\mu_{\hat{r}_k}(t) \neq 0$ for all k $\in$ {1, 2, ..., s}. From (3.23) and (3.24), we may write

$$\mu_{\hat{r}_i}(t) = \max_{t_1 \in \mathbf{D}} \{\mu_r(t), \mu_r(t_1) \mid t_1[R_i] = t[R_i]\} \tag{6.4}$$

Since $\mu_r(t) = 0$, (6.4) implies that for $\mu_{\hat{r}_i}(t) \neq 0$, r must contain at least another tuple t_1 with $t[R_i] = t_1[R_i]$, and $t[R - R_i] \neq t_1[R - R_i]$. But in that case, $\hat{r} = r_1 \bowtie r_2 \bowtie \cdots \bowtie r_s$ will contain both the tuples t and t_1 with nonzero membership values. Simultaneous presence of both t and t_1 in $\hat{r}$ would, however, violate the **rffd**: $R_i \rightsquigarrow R$, contradicting the conclusion that natural join preserves **ffds** (see Lemma 5.2). Therefore, by (6.1), the decomposition ρ has a lossless join. □

Theorem 6.2 suggests that to test lossless join of a decomposition ρ with a set of **rffds** F, we can use the chase process where the tableau transformation rule due to an **rffd** is the same as that due to an **fd**. Thus, suppose that in a tableau **T**, the rows w_1 and w_2 have identical entries in the columns corresponding to the attributes in X, i.e., $w_1[X] = w_2[X]$. Then the transformation rule due to the **rffd** X $\rightsquigarrow$ A, will produce a new tableau $\mathbf{T}_1$ where the entries in w_1 and w_2 at the column corresponding to A also match. While making $w_1[A] = w_2[A]$, if either of them was a distinguished variable in **T**, then the other one is renamed to the same distinguished variable. In case both were nondistinguished, the one with a larger subscript is replaced by the one with the smaller subscript [1, 29, 30, 46]. Applying the **rffd** transformation rule we can now find $\mathbf{T}^* = \mathbf{CHASE}_F(\mathbf{T}_\rho)$, and by Theorem 6.2, ρ has lossless join if $\mathbf{T}^*$ has a row with only distinguished entries.

Example 6.5. Consider the relation ORDER(S, I, Q, P, T), where S = Supplier, I = Item, Q = Quantity, P = Price, and T = Sales tax. Suppose the following resemblance relations are used to compare the domain values.

(1) $\mu_{EQ}(a, b) = 0$ for $a \neq b$, when a, b $\in$ dom(S) or dom(I), i.e., supplier name or item name must exactly match to qualify for equality.

(2) $\mu_{EQ}(a, b) = 1/(1 + \beta \mid a - b \mid)$,

$$\begin{aligned} \text{where } \beta &= 1/100 && \text{for a, b} \in \text{dom(Q)},\\ &= 1/1000 && \text{for a, b} \in \text{dom(P)},\\ &= 1/50 && \text{for a, b} \in \text{dom(T)}. \end{aligned}$$

If required, the modified membership function of "more or less equal" (MLEQ) can be computed as in (5.5). It can be verified that μ_{EQ} or μ_{MLEQ} defined as above satisfies (5.1) and (5.1a).

These resemblance relations are used in the following fuzzy functional dependencies F to be satisfied by ORDER:

SI $\rightsquigarrow$ Q: Supplier and Item determines Quantity.

IQ $\rightsquigarrow$ P: For any Item, Quantity more or less determines Price.

P $\rightsquigarrow$ T: Price determines Sales tax.

Since the resemblance relations satisfy (5.1a), these **ffds** are actually restricted fuzzy functional dependencies. A typical instance of ORDER that satisfies these **rffds** is shown in Table XIV.

Table XIV. An Instance r of ORDER

Supplier	Item	Quantity	Price	Sales-tax	μ
Adams	Nut	300	2500	125	0.6
John	Nut	400	3000	150	0.8
Smith	Nut	300	2500	125	0.7
Adams	Bolt	200	1500	80	0.8
John	Bolt	200	1500	80	0.7
Smith	Bolt	400	2500	125	0.9

Let $\rho = \{R_1(SIQ), R_2(IQP), R_3(PT)\}$ be a decomposition of ORDER. To test whether this decomposition is lossless, we construct the tableau

$$\mathbf{T}_\rho = \begin{array}{c|ccccc} & S & I & Q & P & T \\ \hline R_1 & a_1 & a_2 & a_3 & b_{14} & b_{15} \\ R_2 & b_{21} & a_2 & a_3 & a_4 & b_{25} \\ R_3 & b_{31} & b_{32} & b_{33} & a_4 & a_5 \end{array}$$

Applying **rffd** transformation rule to $\mathbf{T}_\rho$, we obtain

$$\mathbf{CHASE}_F(\mathbf{T}_\rho) = \begin{array}{c|ccccc} & S & I & Q & P & T \\ \hline R_1 & a_1 & a_2 & a_3 & a_4 & a_5 \\ R_2 & b_{21} & a_2 & a_3 & a_4 & a_5 \\ R_3 & b_{31} & b_{32} & b_{33} & a_4 & a_5 \end{array}$$

Here the first row of $\mathbf{CHASE}_F(\mathbf{T}_\rho)$ contains only distinguished variables. Therefore, by Theorem 6.2, the decomposition ρ is lossless join. The projections of r on ρ are shown in Table XV(a, b, and c). The join of these projections can easily be seen to be equal to r. In this case, the decomposition ρ also preserves the given dependencies, i.e., ρ can be considered as an information preserving decomposition [29].

Lastly, we would like to point out that Theorem 6.2 provides only a sufficient condition for the lossless join decomposition of fuzzy relations with **rffds**. Since the **ffd** axioms FF1–FF6 are not complete for all **rffds**, as in Example 6.4, a decomposition ρ of a fuzzy relation with a set F of **rffds** may have a lossless join even though $\mathbf{T}^* = \mathbf{CHASE}_F(\mathbf{T}_\rho)$ has no row with all distinguished variables. However, when EQUAL also satisfies (5.14), i.e., the **ffd** axioms are complete for F (by Theorem 5.1), the absence of any row with all distinguished variables in $\mathbf{T}^*$ can be shown to imply that ρ does not have a lossless join.

Based on these observations, it follows that all the results of the design theory of classical relations with functional dependencies can be directly applied to fuzzy relations with a class of fuzzy functional dependencies where EQUAL is restricted by (5.1a) and (5.14). Since it is not difficult to select a resemblance relation that satisfies these two conditions, the **ffds** belonging to this class can adequately capture the semantics of fuzzy integrity constraints in many real-world applications. In addition, one can now define normal forms of fuzzy relations, or lossless join decomposition of relation schemes which also preserves the given **ffds**, i.e., an information preserving decomposition [29].

Table XV. Projections of an Instance of ORDER on Relation Schemes $R_1(SIQ)$, $R_2(IQP)$, and $R_3(PT)$

Supplier	Item	Quantity	μ
Adams	Nut	300	0.6
John	Nut	400	0.8
Smith	Nut	300	0.7
Adams	Bolt	200	0.8
John	Bolt	200	0.7
Smith	Bolt	400	0.9

(a)

Item	Quantity	Price	μ
Nut	300	2500	0.7
Nut	400	3000	0.8
Bolt	200	1500	0.8
Bolt	400	2500	0.9

(b)

Price	Sales tax	μ
1500	80	0.8
2500	125	0.9
3000	150	0.8

(c)

7. CONCLUSIONS

This paper deals with fuzzy relational data models, with an objective to provide a generalized approach for treating precise, as well as imprecise, data. By selecting suitable interpretations for the membership values, a fuzzy relational model is capable of representing ambiguities in the data values as well as impreciseness in the association among the entities of the database. Since one of the major objectives of fuzzy logic is to represent approximate reasoning used in natural languages, it is expected that in a database environment, appropriate blending of a relational data model and fuzzy logic will enhance the capabilities of the existing database systems. A brief survey of some of the existing proposals for using fuzzy logic in a relational database environment has also been presented.

For a successful blending of fuzzy set theory and relational databases, it is, however, essential to develop a suitable design technique for such systems. In fact, it will be ideal if we can extend some of the widely investigated results in the classical relational database literature. In this quest, we have examined the properties of relational operators, especially projection and join of fuzzy relations. While attempting to extend the data dependencies in classical relations, it was observed that for comparing domain values, a suitable fuzzy measure such as EQUAL becomes useful. Treating "equality" as a fuzzy resemblance relation, a simple and natural extension of classical functional dependency, called *fuzzy functional dependency*, has been proposed. It has been shown that a fuzzy functional dependency can successfully represent integrity constraints that arise out of approximate equality of domain values. In spite of this generalization, the inference axioms for **ffd**s are similar to Armstrong's axioms for classical **fd**s. We have also been able to establish the completeness of these axioms for a class of

ffds by imposing a simple restriction on the resemblance relations. This apparent similarity of the inference axioms is especially useful because many well-known algorithms in the classical **fd** literature, such as Beeri and Bernstein's algorithm [4], can be applied to the **ffds** to perform similar tasks.

In order to obtain adequate decomposition of relation schemes, we next examined the lossless join problem of fuzzy relations in the presence of **ffds**. It is observed that to achieve lossless synthesis of relation schemes, the resemblance relation EQUAL needs to be restricted further. After introducing suitable restrictions on EQUAL, it is shown that a class of **ffds** behaves exactly the same as functional dependencies in classical relations. In fact, the entire design theory of classical relations with functional dependencies becomes applicable to fuzzy relations with this class of fuzzy functional dependencies. Thus one can apply the ABU algorithm to test lossless join decomposition of relation schemes with such **ffds** and can define an information-preserving decomposition or normal forms of fuzzy relations.

It should be mentioned that this work is not a complete nor a conclusive exposition of the capabilities of the fuzzy relations in capturing semantics of the data. However, within the proposed framework, it is not only possible to extend other types of data dependencies, such as multivalued dependency [21], template dependency [43], tuple or equality-generating dependency [6, 24, 29], etc., but one can also model a wider class of integrity constraints. In this paper, we have restricted ourselves to a subclass of fuzzy calculus where truth value of a fuzzy proposition takes binary values. Further generalization of fuzzy integrity constraints can be achieved by using truth-qualified fuzzy propositions [20, 53–55].

ACKNOWLEDGMENTS

The authors are grateful to the referees and to Dr. P. A. Bernstein for the many useful comments and suggestions they made on the first version of this paper. The second author wishes to acknowledge the support provided by the University of Guelph and the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

1. AHO, A. H., BEERI, C., AND ULLMAN, J. D. The theory of joins in relational databases. *ACM Trans. Database Syst. 4*, 3 (Sept. 1979), 297–314.
2. ANVARI, M., AND ROSE, G. F. Fuzzy relational databases. In *Proceedings of the 1st International Conference on Fuzzy Information Processing* (Kuaui, Hawaii, 1984) J. C. Bezdek, Ed., CRC Press, 1984, B-6-3.
3. BALDWIN, J. F. A fuzzy relational inference language for expert systems. In *Proceedings of the 13th IEEE International Symposium on Multivalued Logic* (Kyoto, Japan, 1983). IEEE, New York, 1983, 416–423.
4. BEERI, C., AND BERNSTEIN, P. A. Computational problems related to the design of normal form relational schemas. *ACM Trans. Database Syst. 4*, 1 (Mar. 1979), 30–59.
5. BEERI, C., FAGIN, R., AND HOWARD, J. H. A complete axiomatization for functional and multivalued dependencies in database relations. In *Proceedings of ACM SIGMOD International Conference on Management of Data* (Toronto, 1977). ACM, New York, 1983, 47–61.
6. BEERI, C., AND VARDI, M. Y. Proof procedure for data dependencies. *J. ACM 29*, 4 (Oct. 1982), 986–997.
7. BISKUP, J. A foundation of Codd's relational maybe-operations. *ACM Trans. Database Syst. 8*, 4 (Dec. 1983), 608–636.

8. BRODIE, M. L. On the development of data models. In *On Conceptual Modelling*, M. L. Brodie, J. Mylopoulos, and J. W. Schmidt, Eds. Springer-Verlag, New York, 1984, 19–47.
9. BUCKLES, B. P., AND PETRY, F. E. A fuzzy representation for relational databases. *Fuzzy Sets Syst. 7* (1982), 213–226.
10. BUCKLES, B. P., AND PETRY, F. E. Fuzzy databases and their applications. In *Fuzzy Information and Decision Processes*, M. M. Gupta and E. Sanchez, Eds. North-Holland, Amsterdam, 1982, 361–371.
11. BUCKLES, B. P., AND PETRY, F. E. Query languages for fuzzy databases. In *Management Decision Support Systems Using Fuzzy Sets and Possibility Theory*, J. Kacprzyk and R. R. Yager, Eds. Verlag TUV Rhineland, Cologne, 1985, 241-252.
12. BUCKLES, B. P., AND PETRY, F. E. Uncertainty models in information and database systems. *Inf. Sci. J. 11* (1985), 77–87.
13. BUCKLES, B. P., PETRY, F. E., AND SACHAR, H. S. Design of similarity-based relational databases. Private communication, Nov. 1986.
14. CHANG, S. K., AND KE, J. S. Database skeleton and its application to fuzzy query translation. *IEEE Trans. Softw. Eng. SE-4* (1978), 31–43.
15. CHANG, S. K., AND KE, J. S. Translation of fuzzy queries for relational database systems. *IEEE Trans. Patt. Anal. Mach. Intell. PAMI-1* (1979), 281–294.
16. CODD, E. F. A relational model of data for large shared data banks. *Commun. ACM 13*, 6 (June 1970), 377–387.
17. CODD, E. F. Further normalization of the database relational model. In *Database Systems*, Courant Computer Science Symposia 6, R. Rustin, Ed. Prentice-Hall, Englewood Cliffs, N.J., 1971, 65–98.
18. CODD, E. F. Extending the database relational model to capture more meaning. *ACM Trans. Database Syst. 4*, 4 (Dec. 1979), 397–434.
19. DATE, C. J. *An Introduction to Data Base Systems* (3rd ed). Addison-Wesley, Reading, Mass., 1981.
20. DUBOIS, D., AND PRADE, H. *Fuzzy Sets and Systems: Theory and Applications*. Academic Press, New York, 1980.
21. FAGIN, R. Multivalued dependencies and new normal form for relational databases. *ACM Trans. Database Syst. 2*, 3 (Sept. 1977), 262–278.
22. FAGIN, R. A normal form for relational databases that is based on domains and keys. *ACM Trans. Database Syst. 6*, 3 (Sept. 1981), 387–415.
23. FUKAMI, S., MIZUMOTO, M., AND TANAKA, K. Some considerations on fuzzy conditional inferences. *Fuzzy Sets Syst. 4* (1980), 243–273.
24. GRANT, J., AND JACOBS, B. E. On the family of generalized dependency constraints. *J. ACM 29*, 2 (Oct. 1982), 986–997.
25. KACPRZYK, J., AND ZIOLKOWSKI, A. Database queries with fuzzy linguistic quantifiers. *IEEE Trans. Syst. Man Cybern. SMC-16*, 3 (May/June 1986), 474–479.
26. KAUFMANN, A. *An Introduction to the Theory of Fuzzy Subsets, vol. 1*. Academic Press, New York, 1975.
27. LIPSKI, W., JR. On semantic issues connected with incomplete information databases. *ACM Trans. Database Syst. 4*, 3 (Sept. 1979), 262–296.
28. LIPSKI, W., JR. On databases with incomplete information. *J. ACM 28*, 1 (Jan. 1981), 41–47.
29. MAIER, D. *The Theory of Relational Databases*. Computer Science Press, Rockville, Md., 1983.
30. MAIER, D., MENDELZON, A. O., AND SAGIV, Y. Testing implications of data dependencies. *ACM Trans. Database Syst. 4*, 4 (Dec. 1979), 455–469.
31. MIZUMOTO, M., FUKUMI, S., AND TANAKA, K. Some methods of fuzzy resoning. In *Advances in Fuzzy Set Theory and Applications*, M. M. Gupta et al., Eds. North-Holland, Amsterdam, 1979, 117–136.
32. MIZUMOTO, M., AND ZIMMERMAN, H. J. Comparison of fuzzy reasoning methods. *Fuzzy Sets Syst. 8* (1982), 253–283.
33. PRADE, H. Lipski's approach to incomplete information databases restated and generalized in the setting of Zadeh's possibility theory. *Inf. Syst. 9*, 1 (1984), 27–42.
34. PRADE, H., AND TESTEMALE, C. Generalizing database relational algebra for the treatment of incomplete or uncertain information and vague queries. *Inf. Sci. 34* (1984), 115–143.

35. RAJU, K. V. S. V. N., AND MAJUMDAR, A. K. Fuzzy functional dependencies in fuzzy relations. In *Proceedings of the 2nd International Conference on Data Engineering* (Los Angeles, Feb. 1986). IEEE, New York, 1986, 312–319.
36. RAJU, K. V. S. V. N., AND MAJUMDAR, A. K. The study of joins in fuzzy relational databases. *Fuzzy Sets Syst. 21*, 1 (1987), 19–34.
37. RAJU, K. V. S. V. N., AND MAJUMDAR, A. K. Design and implementation of a fuzzy relational database system. *Inf. Sci. J.* Submitted for publication.
38. REITER, R. On closed world databases. In *Logic and Data Bases*, H. Gallaire and J. Minker, Eds. Plenum Press, New York, 1978, 55–76.
39. REITER, R. Towards a logical reconstruction of relational database theory. In *On Conceptual Modelling*, M. L. Brodie, J. Mylopoulos, and J. W. Schmidt, Eds. Springer-Verlag, New York, 1984, 191–233.
40. RESCHER, N. *Many-Valued Logic.* McGraw-Hill, New York, 1969.
41. RISSANEN, J. Independent components of relations. *ACM Trans. Database Syst. 2*, 4 (Dec. 1977), 317–325.
42. RUSPINI, E. H. Possibility theory approaches for advanced information systems. *IEEE Comput. 9*, 2 (1982), 83–89.
43. SADRI, F., AND ULLMAN, J. D. A complete axiomatization for large class of dependencies in relational databases. In *ACM Symposium on Theory of Computing.* ACM, New York, 1980, 117–122.
44. STONEBRAKER, M. Adding semantic knowledge to a relational database system. In *On Conceptual Modelling*, M. L. Brodie, J. Mylopoulos, and J. W. Schmidt, Eds. Springer-Verlag, New York, 1984, 333–353.
45. STONEBRAKER, M., WONG, E., KREPS, P., AND HELD, G. The design and implementation of INGRES. *ACM Trans. Database Syst. 1*, 3 (Sept. 1976), 189–222.
46. ULLMAN, J. D. *Principles of Database Systems.* Computer Science Press, Rockville, Md., 1980.
47. UMANO, M. FREEDOM-O: A fuzzy database system. In *Fuzzy Information and Decision Processes*, M. M. Gupta and E. Sanchez, Eds. North-Holland, Amsterdam, 1982, 339–347.
48. UMANO, M. Retrieval from fuzzy database by fuzzy relational algebra. In *Fuzzy Information, Knowledge Representation and Decision Analysis*, E. Sanchez, Ed. Pergamon Press, Oxford, UK, 1984, 1–6.
49. YAGER, R. R. A logical on-line bibliographic search: an application of fuzzy sets. *IEEE Trans. Syst. Man Cybern. SMC-10* (1980), 51–53.
50. ZADEH, L. A. Fuzzy Sets. *Inf. Control 8* (1965), 338–353.
51. ZADEH, L. A. Calculus of fuzzy restrictions. In *Fuzzy Sets and Their Applications to Cognitive and Decision Processes.* Academic Press, New York, 1975, 77–95.
52. ZADEH, L. A. Fuzzy sets as a basis for theory of possibility. *Fuzzy Sets Syst. 1*, 1 (1978), 3–28.
53. ZADEH, L. A. A theory of approximate reasoning. In *Machine Intelligence 9*, J. Hayes, et al., Eds. Ellis Horwood Ltd., Sussex, UK, 1979, 149–194.
54. ZADEH, L. A. PRUF—a meaning representation language for natural language. In *Fuzzy Reasoning and its Applications*, E. H. Mamdani and B. R. Gaines, Eds. Academic Press, New York, 1981, 1–66.
55. ZADEH, L. A. A computational approach to fuzzy quantifiers in natural languages. *Comput. Math. Appl. 9*, 1 (1983), 149–184.
56. ZADEH, L. A. Syllogistic reasoning in fuzzy logic and its application to usuality and reasoning with disposition. *IEEE Trans. Syst. Man Cybern. SMC-15*, 6 (1985), 754–762.
57. ZEMANKOVA-LEECH, M., AND KANDEL, A. *Fuzzy Relational Databases—A Key to Expert Systems.* Verlag TUV, Rhineland, Cologne, 1984.
58. ZVIELI, A., AND CHEN, P. P. Entity-relationship modeling and fuzzy databases. In *Proceedings of the 2nd International Conference on Data Engineering* (Los Angeles, Feb. 1986). IEEE, New York, 1986, 320–327.

Received May 1986; revised December 1986; accepted July 1987