Fuzzy Join Dependency in Fuzzy Relational Databases

P. C. Saxena, and D. K. Tayal

Abstract—The join dependency provides the basis for obtaining lossless join decomposition in a classical relational schema. The existence of Join dependency shows that that the tables always represent the correct data after being joined. Since the classical relational databases cannot handle imprecise data, they were extended to fuzzy relational databases so that uncertain, ambiguous, imprecise and partially known information can also be stored in databases in a formal way. However like classical databases, the fuzzy relational databases also undergoes decomposition during normalization, the issue of joining the decomposed fuzzy relations remains intact. Our effort in the present paper is to emphasize on this issue. In this paper we define fuzzy join dependency in the framework of type-1 fuzzy relational databases & type-2 fuzzy relational databases using the concept of fuzzy equality which is defined using fuzzy functions. We use the fuzzy equi-join operator for computing the fuzzy equality of two attribute values. We also discuss the dependency preservation property on execution of this fuzzy equi- join and derive the necessary condition for the fuzzy functional dependencies to be preserved on joining the decomposed fuzzy relations. We also derive the conditions for fuzzy join dependency to exist in context of both type-1 and type-2 fuzzy relational databases. We find that unlike the classical relational databases even the existence of a trivial join dependency does not ensure lossless join decomposition in type-2 fuzzy relational databases. Finally we derive the conditions for the fuzzy equality to be non zero and the qualification of an attribute for fuzzy key.

Keywords—Fuzzy - equi join, fuzzy functions, fuzzy join dependency, type-1 fuzzy relational database, type-2 fuzzy relational database.

I. INTRODUCTION

THE classical relational databases cannot deal the situation when the data to be populated is imprecise in nature, as these databases were designed primarily for the efficient storage and convenient retrieval of a large amount of precise data. They focus on describing precise information and take care of only well defined and unambiguous data. However in real world applications, the data are often partially known (incomplete) or imprecise for e.g. instead of specifying the age of a boy "Ram" as 21years (or more importantly when we just

have an incomplete information about the exact age of Ram, but it is sure that Ram is a young boy) we may simply say that "Ram is young".

Because of the 1st Normal Form property of a classical relational database, no attribute value may contain the values like "young" or "old". Similarly a table in a relational database can store the data as to which college student is studying which subject, but cannot store the value that up to what degree the college student likes that subject. Analogously, if the college student has opted for multiple subjects but he does not like each of these subjects exactly, the 1st Normal form of a classical relational database will disallow this information to be represented as a value of a tuple. Also, sometimes a relation itself is a collection of such tuples which belong to it partially. For an instance if we are interested in storing the information about the "Endangered Species", some of which are "partially endangered", "less endangered", "more endangered" etc, the classical relational data model will not solve the purpose. Thus the representational capability of a relational database simply fails in representing imprecise (fuzzy) data and therefore the fuzzy logic based extensions to database management systems were proposed.

However, the relational database theory is intimately connected to the study of data dependencies (like functional dependency, multi-valued dependency & join dependency etc.) as the data dependencies represent the constraints on data and therefore they must satisfy every relational state of the database. The concept of functional dependency in relational databases was thus extended to fuzzy functional dependency [1, 2] as this dependency was believed to capture a large portion of semantics of the real world. Further since the functional dependencies were not able to represent all the constraints, multi-valued dependency was also extended in fuzzy framework and some authors [3-7] proposed their version of fuzzy multi-valued dependency. However the join dependency still represents a more powerful constraint. The set of join dependencies associated with a relational schema R determines by definition exactly those database schemas that can represent a relation schema R without loss of information [8, 9]. Therefore the join dependency also needs to be extended to the fuzzy framework. Since normalizing a relational schema necessarily needs the decomposition, therefore the concept of lossless join decomposition is crucial, which gives rise to existence of join dependency. Because of

the great importance of join dependency in relational databases, it seems appropriate to study in fuzzy framework the properties of these join dependencies in isolation from other dependencies and thus gives the motivation for this paper. The objective of this paper is to extend the concept of join dependency to the framework of type-1 and type-2 fuzzy relational databases and discuss the results holding in the fuzzy framework.

The paper is organized as follows: Section I introduces the preliminaries and basic notions. In Section II, we briefly discuss the most common fuzzy relational databases available in the literature and introduce type-1 and type-2 fuzzy relational databases which we deal in this paper. In section III we introduce the concept of fuzzy functions and the fuzzy equality based upon it. Using the definition of fuzzy equality we discuss the fuzzy functional dependency (ffd) in type-1 & type-2 fuzzy relational databases. Section IV and V briefly discuss the fuzzy Projection & fuzzy join operators which are used to define the fuzzy join dependency. In section VI we study the conditions for dependency preservation property of the ffd. Finally in section VII the fuzzy join dependency is introduced and the conditions for lossless join decomposition are proposed.

II. FUZZY RELATIONAL DATABASES

Fuzzy databases were developed to capture various types of imprecise information occurring in the real world. They extend the classical databases in two areas: (a) for storing and updating information that is imprecise in nature (b) for processing the imprecise queries. Moreover, the imprecise values in a database system can be broadly classified into two types: (1) imprecise attribute values in a tuple (2) partial membership of a tuple in a relation. Fuzzy databases can very well incorporate both of these of impreciseness. Out of the various approaches proposed in the fuzzy database literature to represent impreciseness in attribute values, we now discuss some of the significant ones. Broadly two approaches have been most popular. First is the similarity based approach, which characterizes the impreciseness by using linguistic terms e.g. poor, fair, good etc. and the degree of similarity between a pair of linguistic terms is characterized by a similarity matrix. However in similarity relations some aspects of max-min transitivity were observed to cause difficulty in modeling the relationship between the domain elements [11]. Even though an extension was added to it to handle fuzzy numbers, the similarity based approach still heavily depends on discrete values. The fuzzy relational model was extended by replacing the similarity matrix with proximity relations (non-transitive) on scalar domains. Raju & Majumdar have also proposed resemblance relations based models treated in [1, 4].

The second approach is based on Zadeh's possibility theory [3] which uses possibility distribution as a value of an attribute to capture the impreciseness of the first type. The second type of impreciseness i.e. the partial membership of a tuple in a relation allows a tuple to be a partial member of a relation, for e.g. the animals which are considered "somewhat endangered" are partial members of the **Endangered-Species** relation. A tuple with a partial membership in a relation is referred as weighted tuple [1, 10, 11].

The possibility based approach described above is believed to be more general and popular; the important reason being that it handles all types of imprecise information. We now provide the basic notations and concepts in two possibility based fuzzy relational data models which we treat in this paper. The models are known as type-1 & type-2 fuzzy relational data models [10].While introducing them; we will show how the two models capture all types of impreciseness in a better way.

Let $U_1, U_2 .... U_n$ be n universes, a fuzzy relation r is a fuzzy subset of $U_1 \times U_2 .... \times U_n$ and is characterized by the n-variant membership function $\mu_r : U_1 \times U_2 .... \times U_n \to [0,1]$. Where μ_r denotes the membership grade of the tuple and takes the values between 0 and 1.

Adhering to the notations of classical relational database theory, a relation schema R in fuzzy relational data model is defined as a finite set of attribute names $\{ A_1, A_2 .... A_n \}$ and is denoted as $R(A_1, A_2 .... A_n)$ or simply by R. An instance of relation R is referred as r. Corresponding to each attribute name $A_i, 1 \le i \le n$, is a set dom(A_i), called the domain of A_i, however the domain of A_i may be a fuzzy set or a set of fuzzy subsets. In the database theory, a fuzzy relation r on a relation scheme $R(A_1, A_2 .... A_n)$ is defined to be a fuzzy subset of $dom(A_1) \times dom(A_2) .... \times dom(A_n)$. Depending on the complexity of dom(A_i), i =1,2..n, the fuzzy relational data model are classified into two categories[10]—

type-1 fuzzy relational data model
type-2 fuzzy relational data model

Similar to the classical relations, a fuzzy relation r is represented as a table with an additional column for $\mu_r(t)$ denoting the membership value of the tuple t in r. As a notational convenience, only those tuples which have $\mu_r(t) > 0$ are shown in the table and for all the tuples missing in the table, it is obvious that $\mu_r(t) = 0$.

A. Type-1 Fuzzy Relational Data Model

A type-1 fuzzy relation may be considered as a first level extension of classical relations, where we are able to capture impreciseness in the association among the entities. In type-1 fuzzy relations, dom (A_i) may be a classical subset or a fuzzy subset of U_i, e.g. consider a relational schema R(N,J,X,S)

of *"highly experienced"* and *"high salaried"* employees in the appropriate universes[1]. Assume that the universe of discourse U_X for the experience is the set of integers is in the range 0-30 and U_S, the universe of discourse of the salary is the set of integers in the range 10,000-1,00,000, then an instance of highly salaried employees can be given as:

Name	Job	Experience	Salary	μ
John	Engineer	8	60,000	0.67
Ashok	Manager	9	70,000	0.80

The above relation shows that the possibility of occurrence of an employee namely John who is an Engineer, having an experience of 8 years and drawing a salary of Rs. 60,000 is 0.67 on a scale of 0 to 1. Similar interpretation is valid for other tuples.

In general for any attribute A_i of a relation schema, let the membership function of $dom(A_i)$ be denoted by μ_{A_i}, for i=1,2…n ; then $dom(A_1) \times dom(A_2)............. \times dom(A_n)$ is a fuzzy subset of $U = U_1 \times U_2.............. \times U_n$ .Hence a type-1 fuzzy relation r is also a fuzzy subset of U with the membership function μ_r .Also for all $(u_1, u_2.........u_n) \in U$, μ_r satisfies

$$\mu_r(u_1, u_2.........u_n) \leq min((\mu_{A_1}(u_1), \mu_{A_2}(u_2)........ ...\mu_{A_n}(u_n)) \qquad (1)$$

In other words $\mu_r(u_1, u_2.........u_n)$ is a fuzzy measure of association among a set of domain values $\{u_1, u_2.........u_n\}$ for sake of explanation, let us take in the above example

$U_1 = dom(A_1) = Name$, $U_2 = dom(A_2) = Job$ and

$U_3 = dom(A_3) = Experience$, $U_4 = dom(A_4) = salary$

Also suppose that the membership function of "high-experience" is defined as

$$\mu_{High-Expereince}(x) = (1+ | x - 10 | /4)^{-1} \quad for\ x \leq 10$$
$$= 1 \quad for\ x > 10$$

and the membership function of "High salary" is defined as

$$\mu_{High-salary}(x) = (1+ | s - 60000 | /20000)^{-1} \quad for\ s \leq 60000$$
$$= 1 \quad for\ s > 60000$$

Let u_1= John, u_2=Engineer, u_3=8, u_4=60000 then (1) gives:

$\mu_{A_1}(u_1) = 1$ (As John is a crisp member)

$\mu_{A_2}(u_2) = 1$ (As Engineer is a crisp member of Job)

$\mu_{A_3}(u_3) = \mu_{High-Experience}(8)$

$$= (1+ | 8 - 10 | / 4)^{-1} = (1 + 2/4)^{-1} = 2/3 = .67$$
$$\mu_{A_4}(u_4) = \mu_{High-Salary}(60000)$$
$$= (1+ | 60000 - 60000 | / 20000)^{-1} = 1$$

Hence membership grade according to (1) is expressed as min (1, 1, .67, 1) = .67

Therefore the tuple (John, Engineer, Experience, salary) in inserted in the table with a membership values of the tuple as 0.67. Similarly the other tuples of the relations are inserted with their respective membership grades.

B. Type –2 Fuzzy Relational Data Model

Although the type-1 fuzzy relations enable us to represent impreciseness in the association among the data values, its role in capturing the uncertainty in data values is rather limited [10]. For e.g. in a type-1 fuzzy relational model for Employee viz R(Name, Job, Experience, Salary) one is not permitted to specify the salary of John to be in range $30,000-40,000 or experience of some employee Jack to be "low". So what about answering queries with imprecise values? The type-2 fuzzy relations provides further generalization by allowing $dom(A_i)$ to be even a set of fuzzy sets. By enlarging $dom(A_i)$, type-2 relations represent a wider type of impreciseness in data values which can be considered a second level generalizations of classical relations. As in type–2 fuzzy relational data model , for any attribute A_i, $dom(A_i)$ may be considered as a set of fuzzy sets in U_i, therefore a tuple $t = (a_1, a_2......... a_n)$ in $\mathbf{D} = dom(A_1) \times dom(A_2)............. \times dom(A_n)$ becomes a fuzzy subset of $U = U_1 \times U_2.............. \times U_n$ with,

$$\mu_t(u_1, u_2.........u_n) = min [(\mu_{a_1}(u_1), \mu_{a_2}(u_2)...........\mu_{a_n}(u_n)] \qquad (2)$$

where $u_i \in U_i$, for i=1,2…..n. Since (2) holds for all $u_i \in U_i$, i=1, 2…..n and using the definition of a fuzzy relation, a type-2 fuzzy relation r is characterized as a fuzzy subset of **D.** On applying the definition of fuzzy subset, the membership function $\mu_r : D \rightarrow [0,1]$ must satisfy the following condition:

$$\mu_r(t) \leq \max_{(u_1, u_2....u_n)} [min\{\mu_{a_1}(u_1), \mu_{a_2}(u_2)..... .... \mu_{a_n}(u_n)\} \qquad (3)$$

Where $t = (a_1, a_2.........a_n) \in \mathbf{D}$

Example 1: Consider a fuzzy relation schema [9] EMPLOYEE(Name, Age, Department, salary, Expertise-Domain) in a UNIVERSITY RESEARCH CENTRE (Table I), which contains the information about name, age, the department where an employee works, his status of salary and his areas of expertise. In the EMPLOYEE relation,

dom(Name) and dom(department) are assumed to be crisp sets while dom(Age), dom(salary) and dom(Expertise-Domain) are sets of fuzzy sets in the universes U_{Age}, U_{salary} and U_{ED}. The first tuple specifies that possibility of a person namely "Dass" aged 56 years working in Maths department, having low salary with areas of expertise including Vector Analysis, Modern Algebra and Linear Algebra up to the level of 0.7, 0.8 and 0.6 respectively is 0.7. Similar interpretations can be made for other tuples and the membership functions can be given by:

$$\mu_{high}(y) = (1 + \frac{1}{20,000}|y - 60000|)^{-1} \quad \text{for } y \leq 60,000$$

$$= 1 \qquad\qquad \text{for } y > 60,000$$

$$= 1 - \mu_{low}(y) \qquad\qquad (4)$$

, where $y \in U_{salary}$

TABLE I
EMPLOYEE

	Name	Age	Department	Expertise Domain	μ
t_1	Dass	56	Maths	{0.7/Vector Analysis,.8/Modern Algebra,.6/Linear Algebra}	0.7
t_2	Jain	{0.7/53, 1/54, .08/55}	Mgmt	Risk Management	0.9
t_3	Arya	{.05/54, 0.5/55}	Comp Engg	Networking	0.6
t_4	Roy	58	Physical sciences	{0.8/Magnetism, 0.7/Fluid Dynamics}	0.7

In Table I, it can be noted that the values of the attribute in a tuple contains fuzzy sets like "High" (Salary). The important issue here is of computing the equality of two attributes values since unlike classical databases equality cannot be directly computed in fuzzy databases. In fuzzy databases a comparison of two fuzzy sets is required to compute the fuzzy equality, so it becomes essential to devise a mechanism to evaluate the fuzzy equality between two fuzzy sets. In this paper we will use the fuzzy equality based on fuzzy functions introduced in section III.

III. FUZZY FUNCTIONS AND FUZZY EQUALITY

The representation of data dependencies in relational database model needs comparison of the attribute values of a relation and fuzzy databases are no exception. In this section we devise a methodology to compute the fuzzy equality of two attribute values to compute the equality of attribute values occurring in a type-2 fuzzy relational schema. This methodology holds well for a type-1 fuzzy relational model also.

We find that the equality measures like resemblance relations [1, 2, 15], and conformance [7] etc. are highly dependent on the semantics of the attribute values, the important reason being that they are defined differently for every attribute domain. For instance if a fuzzy relation has 'n' attributes, then 'n' different resemblance relations have to be defined.

On the other side, in this paper we will be using a general measure of the fuzzy equality based on the concept of fuzzy function [18,19].This approach does not require the computation of different resemblance relations /conformance measure for each attribute domain. The approach followed by us in this paper will thus relieve the database designer from indulging in the hectic and error prone process of defining these equality measures, which is sometimes controversial also. We now introduce the concept of fuzzy functions and the fuzzy equality based upon it.

Let A and B be two sets. A binary relation f on A × B is a fuzzy subset of A × B, characterized by the membership function $\mu_f : A \times B \to [0, 1]$. Let E_A and E_B denote the fuzzy equality on A and B respectively. Then f is called as a fuzzy function from A to B [18], if

(F1) for each $x \in A$, there exists a $y \in A$ such that $\mu_f(x, y) > 0$

(F2) for all $x_1, x_2 \in A$ and for all $y_1, y_2 \in B$,

$$\mu_f(x_1, y_1) \wedge \mu_f(x_2, y_2) \wedge E_A(x_1, x_2) \leq E_A(y_1, y_2)$$

Again suppose $A = A_1 \times .... A_n$ and $B = B_1 \times ....B_m$ where each A_i, i=1…..n and B_j, j=1…..m are sets and f: $X \to Y$ is a function (as we will be considering the set [0, 1] to be a totally ordered set throughout this paper so that the infimum i.e. $\wedge$ can be taken to be equivalent to 'minimum') then the fuzzy equality on A is defined in a manner

$$E_A(x, y) = \min_{i \in [1,n]} \{E_{A_i}(x_i, y_i)\}$$

For all $(x,y) \in A \times A$,where E_{A_i} is a fuzzy equality on A_i. E_B can be defined in a similar manner.

To obtain the fuzzy equality over the tuples or the projection over the tuples, the above notion of fuzzy equality is generalized to Cartesian product of sets as follows:

If $A = A_1 \times .... A_n$ where each A_i, i =1,2…n is a set , then $X = (x_1, x_2......x_n)$ and $Y = (y_1, y_2......y_n)$ are equal iff $x_i = y_i$, i=1,2….n that is X=Y, for all i=1,2….n . Accordingly the corresponding graded function,

$E_A : A \times A \to [0, 1]$ can be defined as

E4) $E_A(x, y) = \min_{i \in [1,n]} \{E_{A_i}(x_i, y_i)\}$

Denoting the interval [0, 1] by I and all the fuzzy subsets of a crisp set by I^X ,

Let $I(A, B) = \{(x \in X : \mu_A(x) \leq \mu_B(x)\}$ be a crisp set, for all A, B $\in I^X$ and denoting its characteristic function by $\chi_{I(A,B)}$,we observe that

$$A \subseteq B \Leftrightarrow \bigwedge_{x \in X} \{\chi_{I(A,B)}(x)\} = 1$$

$$A \nsubseteq B \Leftrightarrow \bigwedge_{x \in X} \{\chi_{I(A,B)}(x) = 0$$

Demirci [18] defined the mappings $\widetilde{\subseteq}_*, \cong_* : I^X \times I^X \to I$

as follows

$$[\widetilde{\subseteq}_* (A,B)\}] = [\widetilde{\subseteq} (A,B)] \vee (\bigwedge_{x \in X} \{\chi_{I(A,B)}(x)\})$$

$$[\cong_* (A,B)] = [\widetilde{\subseteq}_* (A,B)] \wedge [\widetilde{\subseteq}_* (B,A)]$$

where

$$[\widetilde{\subseteq} (A,B)] = \bigwedge_{x \in X} \{(A^c \vee B)(x)\}$$

and A^c is the complement of fuzzy set A. μ_A is the membership function of a set A and $\mu_{A^c}(x) = 1 - \mu_A(x)$, for all $x \in X$.

The real number $[\cong_* (A,B)] = [\widetilde{\subseteq}_* (A,B)] \wedge [\widetilde{\subseteq}_* (B,A)]$; $A, B \in I^X$ shows the degree of equality to which a fuzzy set A is equal to a fuzzy set B. Thus the mapping $\cong_* : I^X \times I^X \rightarrow I$ is a fuzzy equality on I^X. Consequently definition of fuzzy equality is extended to tuples of fuzzy sets as follows: Let X_i, $i = 1, 2, \ldots n$ be sets and A_i, B_i be fuzzy sets in I^{X_i}. Let the fuzzy equality on I^{X_i} be $E_{I^{X_i}}$ that is $[E_{I^{X_i}}(A_i, B_i)] = [\widetilde{\subseteq}_* (A,B)] \wedge [\widetilde{\subseteq}_* (B,A)]$.

Similarly the fuzzy equality $E_{I^X} : I^X \times I^X \rightarrow [0, 1]$, where $I^X = I^{X_1} \times I^{X_2} \times \ldots\ldots\ldots I^{X_n}$ is defined as:

$$E_{I^X}((A_1, A_2 \ldots A_n), (B_1, B_2 \ldots B_n)) = \bigwedge_i E_{I^{X_i}}(A_i, B_i) \text{ for all}$$

$(A_1, A_2 \ldots A_n), (B_1, B_2 \ldots B_n) \in I^X$.

Thus we can say that if $R = \{A_1, A_2 \ldots A_n\}$ be a relation scheme, a fuzzy relation r over R is a fuzzy subset of $\text{dom} A_1 \times \text{dom} A_2 \times \ldots \text{dom} A_n$. The membership function of r is denoted as μ_r.

A tuple $t = (a_1, a_2 \ldots a_n)$ in $\text{dom} A_1 \times \text{dom} A_2 \times \ldots \text{dom} A_n$ is thus associated with the grade of membership $\mu_r(t)$ in r. i.e. $r = \{\mu_r(t)/t : t \in \text{dom} A_1 \times \text{dom} A_2 \times \ldots \text{dom} A_n\}$. Further let $(i_1, i_2 \ldots i_k)$ and $(j_1, j_2 \ldots j_m)$ denote two subsequences of $(1, 2 \ldots n)$ and let $X = \{A_{i_1}, A_{i_2} \ldots A_{i_k}\}$, $Y = \{A_{j_1}, A_{j_2} \ldots A_{j_m}\}$, $\text{dom} X = \text{dom} A_{i_1} \times \text{dom} A_{i_2} \ldots \times \text{dom} A_{i_k}$ and $\text{dom} Y = \text{dom} A_{j_1} \times \text{dom} A_{j_2} \ldots \times \text{dom} A_{j_m}$; then the fuzzy relation r induces the binary fuzzy relation over XY, which is a fuzzy subset of $\text{dom} X \times \text{dom} Y$ given by

$$r_{XY} = \{\mu_r(t, t')/(t, t') : (t, t') \in \text{dom} X \times \text{dom} Y\} \text{ and}$$

$$\mu_r(t, t') = \inf_{t''} \{\mu_r(t'') : t''[X] = t \text{ and } t''[Y] = t'\}$$

It implies that whenever there is a set of tuples t'' (i.e. more than one tuple is present) such that $t''[x] = t$ and $t''[y] = t'$, where $t \in \text{dom} X$ and $t' \in \text{dom} Y$, then the infimum over all such t'' must be taken to compute the grade $\mu_r(t, t')$.

We simplify the notation and denote fuzzy equality as defined above on $\text{dom} A_i$, for an attribute A_i, $i = 1, 2 \ldots n$ by E_{A_i} and not by $E_{\text{dom} A_i}$. In this paper we will be using the symbols $\cong_*$ & E_{A_i} interchangeably to denote fuzzy equality over an attribute A_i. Similarly when $X = \{A_{i_1}, A_{i_2} \ldots A_{i_k}\}$ is a subset of R, a fuzzy equality on $\text{dom} A_{i_1} \times \text{dom} A_{i_2} \ldots \text{dom} A_{i_k}$ is denoted by E_X.

Example 2: Let us consider an example similar to as in [20]. Let $R = \{A, B, C, D\}$ be a type -2 fuzzy relation scheme and r be a fuzzy relation over R containing three tuples t_1, t_2, t_3 defined as follows:

	A	B	D	μ
t_1	a	b_1	d_1	0.5
t_2	a	b_2	d_1	0.6
t_3	a	b_3	d_2	0.35

where A is a classical set and $b_1, b_2, b_3 \in I^{\text{domB}}$ are fuzzy sets and $d_1, d_2 \in D$ where D is an ordinary set.

$$\mu_a(x) = \begin{cases} 1 \text{ if } x = u, u \in \text{domA} \\ 0 \text{ if } x \neq u \end{cases}$$

$\therefore E_{\text{domA}}(a, a) = 1$

Let $\text{domB} = \{v_1, v_2\}$, let the membership function of b_1, b_2 and b_3 be

$$\mu_{b_1}(x) = \begin{cases} 1 & \text{if } x = v_1, v_1 \in \text{domB} \\ 0.7 & \text{if } x = v_2, v_2 \in \text{domB} \\ 0 & \text{if } x \neq v_1, x \neq v_2 \end{cases}$$

$$\mu_{b_2}(x) = \begin{cases} 0.6 & \text{if } x = v_1, v_1 \in \text{domB} \\ 1 & \text{if } x = v_2, v_2 \in \text{domB} \\ 0 & \text{if } x \neq v_1, x \neq v_2 \end{cases}$$

$$\mu_{b_3}(x) = \begin{cases} 0.3 & \text{if } x = v_1, v_1 \in \text{domB} \\ 1 & \text{if } x = v_2, v_2 \in \text{domB} \\ 0 & \text{if } x \neq v_1, x \neq v_2 \end{cases}$$

Now for the pair of tuples t_1 and t_2, let us calculate $E_{\text{domB}}(b_1, b_2)$

Then $I(b_1, b_2) = \{x \in \text{domB} : \mu_{b_1}(x) \leq \mu_{b_2}(x)\}$ which is calculated as

For $x = v_1 : \mu_{b_1}(v_1) = 1$ and $\mu_{b_2}(v_1) = 0.6$

So $\mu_{b_1}(x) \nleq \mu_{b_2}(x)$ for $x = v_1$

For $x = v_2 : \mu_{b_1}(v_2) = 0.7$ and $\mu_{b_2}(v_2) = 1$

So $\mu_{b_1}(x) \leq \mu_{b_2}(x)$ for $x = v_2$

For $x \neq v_1, x \neq v_2$: $\mu_{b_1}(x) = \mu_{b_2}(x) = 0$

So $\mu_{b_1}(x) \leq \mu_{b_2}(x)$

Thus $I(b_1, b_2) = \{x$

$= v_2, x \neq v_1$ and $x \neq v_1 : \mu_{b_1}(x) \leq \mu_{b_2}(x)\} = \text{dom } B - \{v_1\}$

Now $[\underset{\sim}{\subseteq}(b_1, b_2)] = \underset{x \in \text{domB}}{\Lambda}\{b_1^c \vee b_2(x)\}$

$= \underset{x \in \text{domB}}{\Lambda}\{(1 - b_1) \vee b_2)(x)\}$

$= \underset{x \in \text{domB}}{\Lambda}\{(1 - \mu_{b_1}(x)) \vee \mu_{b_2}(x)\}$

for $x = v_1$:

$\{(1 - \mu_{b_1}(v_1)) \vee \mu_{b_2}(v_1)\} = \{(1 - 1) \vee 0.6\} = (0 \vee 0.6) = 0.6$

For $\qquad\qquad\qquad\qquad x = v_2$:

$\{(1 - \mu_{b_1}(v_2)) \vee \mu_{b_2}(v_2)\} = \{(1 - 0.7) \vee 1\} = (0.3 \vee 1) = 1$

For $\qquad\qquad\qquad\qquad x \neq v_1, x \neq v_2$:

$\{(1 - \mu_{b_1}(x)) \vee \mu_{b_2}(x)\} = \{(1 - 1) \vee 0\} = 1$

Therefore $[\underset{\sim}{\subseteq}(b_1, b_2)] = 0.6 \wedge 1 \wedge 1 = 0.6$,

Similarly $[\underset{\sim}{\subseteq}(b_2, b_1)] = 0.7$

Now, by the definition of $\chi_{I(b_1, b_2)}$,

$\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_1, b_2)}(x)\} = \begin{cases} 1 & \text{if} \quad b_1 \subseteq b_2 \\ 0 & \text{if} \quad b_1 \not\subseteq b_2 \end{cases}$

Therefore $\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_1, b_2)}(x)\}$

$= \underset{x \in \text{domB}}{\Lambda}\{\chi_{\{x \in \text{domB}: \mu_{b_1}(x) \leq \mu_{b_2}(x)\}}(x)\}$ which is given by

$\chi_{\{x \in \text{domB}: \mu_{b_1}(v_1) \leq \mu_{b_2}(v_1)\}}(x = v_1) \wedge \chi_{\{x \in \text{domB}: \mu_{b_1}(v_2) \leq \mu_{b_2}(v_2)\}}(x = v_2)$

$\wedge \chi_{\{x \neq v_1, x \neq v_2 : \mu_{b_1}(x) \leq \mu_{b_2}(x)\}}(x \neq v_1, x \neq v_2)$

$= \chi_{\{v_1 \in \text{domB}: 1 \geq .6\}}(x = v_1) \wedge \chi_{\{v_2 \in \text{domB}: .6 \leq 1\}}(x = v_2)$

$\wedge \chi_{\{x \neq v_1, x \neq v_2 : \in \text{domB}: 0 \leq 0\}}(x \neq v_1, x = v_2)$

$= 0 \wedge 1 \wedge 1 = 0$

Hence

$[\underset{\sim}{\subseteq}_*(b_1, b_2)] = [\underset{\sim}{\subseteq}(b_1, b_2)] \vee (\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_1, b_2)}(x)\}) = 0.6 \vee 0 = 0.3$

$[\underset{\sim}{\subseteq}_*(b_2, b_1)] = [\underset{\sim}{\subseteq}(b_2, b_1)] \vee (\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_2, b_1)}(x)\}) = 0.7 \vee 0 = 0.7$

So, $E_{\text{domB}}(b_1, b_3) = 0.3 \wedge 0.7 = 0.3$

Similarly $E_{\text{domB}}(b_2, b_3)$ and $E_{\text{domB}}(b_1, b_3)$ can be calculated as follows:

$[\underset{\sim}{\subseteq}(b_2, b_3)] = 0.3$

$[\underset{\sim}{\subseteq}(b_3, b_2)] = 0.6$

And

$\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_2, b_3)}(x)\} = 0$, similarly $\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_3, b_2)}(x)\} = 1$,

$[\underset{\sim}{\subseteq}_*(b_2, b_3)] = [\underset{\sim}{\subseteq}(b_2, b_3)] \vee (\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_2, b_3)}(x)\}) = 0.3 \vee 0 = 0.3$

$[\underset{\sim}{\subseteq}_*(b_3, b_2)] = [\underset{\sim}{\subseteq}(b_3, b_2)] \vee (\underset{x \in \text{domB}}{\Lambda}\{\chi_{I(b_3, b_2)}(x)\}) = 0.6 \vee 1 = 1$

Therefore

$[\cong_*(b_2, b_3)] =$

$E_{\text{domB}}(b_2, b_3) = [\underset{\sim}{\subseteq}_*(b_2, b_3)] \wedge [\underset{\sim}{\subseteq}_*(b_3, b_2)] = 0.3 \wedge 1 = 0.3$

Similarly $[\cong_*(b_1, b_2)] = E_{\text{domB}}(b_1, b_2) = 0.6$

So we conclude that $\cong_*(b_1, b_3) = E_{\text{domB}}(b_1, b_3) = 0.3$

$\cong_*(b_2, b_3) \, E_{\text{domB}}(b_2, b_3) = 0.2$ And

$E_{\text{domB}}(b_1, b_2) = 0.6$

In this way the equality of the fuzzy sets which occur as the value of the attributes can be computed using fuzzy functions. The fuzzy equality of two fuzzy sets will help us in computing the equality of two tuple values for an attribute.

A. Fuzzy Equality of Normal and Subnormal Sets

We have seen in section II that an attribute value of a tuple in a type-2 fuzzy relation schema can be a fuzzy set. The *height* h(A)[11] of a fuzzy set A is the largest membership grade obtained by any element in that set i.e. h(A)= supremum$\{A(x)\}, x \in A$. The fuzzy set A is called normal when h(A) = 1 and it is called subnormal when h(A)< 1 [11]. It is obvious that a normal/subnormal fuzzy set can occur as an attribute value. However subnormal and normal fuzzy sets show different properties in obeying fuzzy equality. Since the fuzzy join attribute may require computing the fuzzy equality of two subnormal/ normal fuzzy sets, we focus on these properties in this section.

Lemma 1: If A and B are two fuzzy subnormal fuzzy sets defined over a set X , the fuzzy equality of A and B cannot be equal to 0 i.e. $[\cong_*(A, B)] > 0$ always .

Proof: Let the set X is given by X= $\{a_1, a_2, a_3, a_4, a_5 ... a_1\}$. Suppose that the two fuzzy sets A and B defined over X are both subnormal.

Let A= $\{m_1/a_1 + m_2/a_2 + m_3/a_3\}$ and B = $\{n_3/a_3 + n_5/a_5\}$, where m_i ; i=1,2,3 denotes the membership of a_i in the fuzzy set A and n_j; j=3,5 denotes the membership of a_j in the fuzzy set B.

It can be easily noted that $0 < m_i, n_j < 1$ (since both the fuzzy sets are subnormal and by convention any member of a fuzzy set with membership value equal to 0 is not shown in the set).

Now using the definition of fuzzy equality $[\cong_*(A, B)] = [\underset{\sim}{\subseteq}_*(A, B)] \wedge [\underset{\sim}{\subseteq}_*(B, A)]$ and thus we conclude that the fuzzy equality of two fuzzy sets A and B can be equal to 0 only when either of $[\underset{\sim}{\subseteq}_*(A, B)]$ or $[\underset{\sim}{\subseteq}_*(B, A)]$ (or both) is equal to 0. We

will prove that none of them can be equal to 0. We will only provide a proof for $[\widetilde{\cong}_* (A,B)] \neq 0$ here, as the proof for $[\widetilde{\cong}_* (B,A)] \neq 0$ can be constructed on similar guidelines.

For proving that $[\widetilde{\cong}_* (A,B)] > 0$, we note that by definition

$$[\widetilde{\cong}_* (A,B)] = [\widetilde{\cong}(A,B)] \vee (\underset{x \in X}{\wedge}\{\chi_{I(A,B)}(x)\}) \qquad (5)$$

so $[\widetilde{\cong}_* (A,B)]$ can only be equal to 0 only when both the $[\widetilde{\cong}(A,B)]$ and $(\underset{x \in X}{\wedge}\{\chi_{I(A,B)}(x)\})$ become equal to 0. Considering $(\underset{x \in X}{\wedge}\{\chi_{I(A,B)}(x)\})$, we find that it can be equal to 0 only when $\exists$ some $x \in X$ such that $\mu_A(x) > \mu_B(x)$. If no such x exists, we have nothing to prove. So let us assume that $\exists$ an element $x' \in X$ such that $\mu_A(x') > \mu_B(x')$ so that in (5), $(\underset{x \in X}{\wedge}\{\chi_{I(A,B)}(x)\}) = 0$.

Now $\forall x \in X$, let us consider the value of $[\widetilde{\cong}(A,B)] = \underset{x \in X}{\wedge}\{(A^c \vee B)(x)\}$. Here two cases arise:

Case 1: $\mu_A(x) = 0 = \mu_B(x)$

Then $(A^c \vee B)(x) = (1-0) \vee 0 = 1$

Case 2: At least one of $\mu_A(x)$ or $\mu_B(x)$ is greater than 0

Let us assume that $\mu_A(x) = m > 0$ and $\mu_B(x) = 0$ so that $(A^c \vee B)(x) = (1-m) \vee 0$, but since m>0 so (1-m) > 0 so that $(A^c \vee B)(x) > 0$. Hence for any value of $x \in X$, we have $(A^c \vee B)(x) > 0$, so that

$$\underset{x \in X}{\wedge}\{(A^c \vee B)(x)\} > 0 \qquad (6)$$

Using (5) and (6) we obtain the result that the fuzzy equality of two subnormal fuzzy sets cannot be equal to 0.

We now obtain the conditions when two non-subnormal fuzzy sets are not fuzzily equal.

Lemma 2: Let A and B be two fuzzy sets defined over the same domain, with at least one of them normal .If any member of the normal fuzzy set having the membership value 1 has the membership value 0 in the other fuzzy set , then $[\cong_* (A,B)] = 0$.

Proof: The proof can easily be obtained using the proof of Lemma 1. In Lemma 1, let $x' \in X$ be any element such that $\mu_A(x') = 1$ and $\mu_B(x') = 0$. Since $\exists x' \in X$ here, such that $(1 = \mu_A(x')) > (\mu_B(x') = 0)$, so $(\underset{x \in X}{\wedge}\{\chi_{I(A,B)}(x)\})$ is necessarily 0. We are thus left with to show that $[\widetilde{\cong}(A,B)] = 0$. Now here it is clear that all the elements belonging to domain of X do not have membership equal to 0 because of existence of $x' \in X$ with $\mu_A(x') = 1$, thus for this element x', we have $(A^c \vee B)(x') = (1-1) \vee 0 = 0 \vee 0 = 0$

Hence $[\widetilde{\cong}(A,B)] = \underset{x \in X}{\wedge}\{(A^c \vee B)(x)\} = 0$. Using the definition of $[\widetilde{\cong}_* (A,B)]$, we have that it comes to be equal to 0, making $[\cong_* (A,B)] = 0$.

B. Fuzzy Equality over Different Domains

Since it is necessary to include different type of domains viz classical sets & fuzzy sets (in type-1 fuzzy relations) and classical sets & set of fuzzy subsets(in type -2 fuzzy relations) to correctly represent the impreciseness inhibited , the issue of equality of two values of tuples corresponding to an attribute is quite crucial . Sometimes we have to compute equality of a crisp value to: a crisp value, a fuzzy set or to even a set of fuzzy subsets. To compute the fuzzy equality over different domains, the authors define the fuzzy equality of two values in [21] as follows.

Definition 1 [21]**:** Let t_1 and t_2 be any two tuples of a fuzzy relation R. Let A be any attribute of R and let $t_1[A]$ and $t_2[A]$ denote the values of the tuples t_1 and t_2 for the attribute A, then fuzzy equality of $t_1[A]$ and $t_2[A]$ can be calculated as: $E_{(t_1[A],t_2[A])}(t_1[A],t_2[A]) =$

$$\begin{cases} = 1 \text{ if dom(A) is an ordinary set and } t_1[A] = t_2[A] \\ = 0 \text{ if dom(A) is an ordinary set and } t_1[A] \neq t_2[A] \\ = 1 \text{ if dom(A) is a fuzzy set and } t_1[A] = t_2[A] \\ = 0 \text{ if dom(A) is a fuzzy set and } t_1[A] \neq t_2[A] \\ \mu_{t_2[A]}(t_1[A]) \text{ if dom (A) is a set of fuzzy subsets} \\ \qquad \text{and } t_1[A] \text{ is crisp value and } t_2[A] \text{ is a fuzzy subset} \\ \mu_{t_1[A]}(t_2[A]) \text{ if dom (A) is a set of fuzzy subsets} \\ \qquad \text{and } t_2[A] \text{ is crisp value and } t_1[A] \text{ is a fuzzy subset} \\ \cong_* (t_1[A],t_2[A]) \text{ if dom (A) is a set of fuzzy subsets} \\ \qquad \text{and both } t_1[A] \text{ and } t_2[A] \text{ are fuzzy subsets} \\ E_A(t_1[A],t_2[A]) \text{ if } t_1[A] \text{ and } t_2[A] \text{ are crisp values} \\ \qquad \text{and dom (A) is a set of fuzzy subsets ;} \\ \text{where } E_A(t_1[A],t_2[A]) \text{ is a domain dependent function} \end{cases}$$

$$(7)$$

Thus, definition 1 allows us to calculate the fuzzy equality of two values occurring in the same domain. It will thus help us in calculating the fuzzy equality of two values which are used in defining the fuzzy functional dependency and fuzzy join operator. However we realize that while obtaining the fuzzy join of two type-2 fuzzy relations, it may be case that although joining attribute from one relation and the joining attribute from another relation may be defined over the same universe, but one is simply a crisp set while another is a set of fuzzy subsets or that while joining two type-1 fuzzy relations one attribute is a crisp set while another attribute is a fuzzy set defined over the same set as the previous attribute. In [21] the authors define the fuzzy join compatibility to resolve the above mentioned issues as follows:

Definition 2 [21]**:** Two set of attributes X($A_1, A_2 \ldots A_n$) and Y($B_1, B_2 \ldots B_n$) belonging to two fuzzy relations (either both type-1 or type-2) are said to be fuzzy- join compatible if

they have the same degree n, and if for $1 \leq i \leq n.$, either $dom(A_i) = dom(B_i)$ or $dom(A_i) =$ fuzzy set over $dom(B_i)$ or $dom(A_i) = I^{dom(B_i)}$; i.e. the set of fuzzy subsets over domain of B_i.

Thus the fuzzy-join compatibility checks the compatibility of the domains of the two sets of attributes X and Y in the fuzzy environment. Thus if we want to join two sets of attributes, then they must have to be fuzzy-join compatible.

C. Fuzzy Functional Dependency Using Fuzzy Equality

The integrity constraints in relational database systems can be broadly classified into two groups [10]--

(1) Domain Dependency – This restricts admissible domain values of attributes, e.g. "age of an employee is less than 62 years" .

(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 design of database systems. The functional dependency is the most common and widely attended data dependency. Some authors [2, 10, 11] have defined functional dependencies in fuzzy relational databases relational database. The authors have defined the fuzzy functional dependency in type-1 & type-2 fuzzy relational databases as follows [20]:

Definition 3 [20]**:** A fuzzy relation r over a relation scheme R = { $A_1, A_2 ...... A_n$ } satisfies a fuzzy functional dependency (ffd) X $\rightarrow$ Y where X, Y $\subseteq$ R, if r_{XY} is a fuzzy partial function i.e. for all $t_1, t_2 \in dom\, X$ with $E_X(t_1, t_2) \neq 0$ and for all $t_1', t_2' \in dom\, Y$

$$\mu_{r[XY]}(t_1, t_1') \wedge \mu_{r[XY]}(t_2, t_2') \wedge E_X(t_1, t_2) \leq E_Y(t_1', t_2') \tag{8}$$

For further elaboration of the fuzzy functional dependency and its examples, [20] may be referred.

IV. FUZZY PROJECTION OPERATOR

The projection operation plays an important role in the theory of lossless join decomposition in classical relational databases. Using the duplicity elimination property of the projection operation, tuples which are "identical" in a fuzzy relation are collected and the tuple with the highest membership grade in the fuzzy relation is kept and the others are discarded. The same role of projection operation is also extended to fuzzy relational databases. Raju & Majumdar[1] define the fuzzy projection as :

Definition 4 [1]**:** Let r be an instance of a fuzzy relation schema $R(A_1, A_2 .... A_n)$, and let $(i_1, i_2 .... i_k)$ be a subsequence of (1, 2...n). The fuzzy projection $r_i = P_{R_i}(r)$ is a

k-ary fuzzy relation in $dom(A_{i_1}) \times dom(A_{i_2}) \times ....$ $\times dom(A_{i_k})$ with the membership function μ_{r_i} given by

$$\mu_{r_i}(t) = \max_{t_r}\{\mu_r(t_r) \mid t_r[R_i] = t\} \tag{9}$$

where t_r is a tuple of r and $t \in dom(A_{i_1}) \times dom(A_{i_2}) \times .... \times dom(A_{i_k})$.

Thus the tuples of r_i are the restrictions of the tuples of r, as in case of classical relations. The max operator 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 case of classical relations, the grades have binary values, and therefore the above condition simply leads to duplicate removal.

V. FUZZY JOIN OPERATOR

The very issue of joining two fuzzy relations has also been treated by few researchers in the past. Raju & Majumdar[1] have defined a cylindrical extension based join–operator in fuzzy relational databases as follows:

Let $r_1, r_2 .... r_s$ are a fuzzy relational database for relational schemes, $R_i = A_{i_1} .... A_{i_{k_i}}$, where i=1,2........s having

membership function μ_{r_i}. Let $\bigcup_{i=1}^{s} R_i = (A_1, A_2 .... A_n)$.

Then the join of fuzzy relations $r_1, r_2 .... r_s$ denoted by $\bar{r}$ is a fuzzy relation over $U_1 \times U_2 .... \times U_n$. The membership function of $\bar{r}$ is defined by

$$\mu_{\bar{r}}(a_1, a_2 .... a_n) = \min[(\mu_{\hat{r}_1}(a_1, a_2 .... a_n), ... \mu_{\hat{r}_s}(a_1, a_2 .... a_n)]$$

, where $\mu_{\hat{r}_i}(a_1, a_2 .... a_n)$ is the cylindrical extension of $\mu_{\hat{r}_i}(a_{i_1} .... a_{i_{k_i}})$.

However the above fuzzy join operator holds well only in case of type-1 fuzzy relational databases as it requires that all the values of the fuzzy relations must be crisp values and none of them can be a subnormal/normal fuzzy set. It does not provide any provision for calculating the fuzzy equality of two fuzzy sets on the basis of which an equi-join can be operated; therefore the cylindrical extension based join operator cannot work for type-2 fuzzy relational databases. To overcome this difficulty the authors have proposed a fuzzy equi-join operator [21] which fits well for both type-1 and type-2 fuzzy relational databases. For an elaboration of the fuzzy equi-join operator [21] may be referred. In the next sections we will use this fuzzy equi-join operator to study the Join dependency and Dependency preservation property in type-1 and type-2 fuzzy relational databases.

VI. DEPENDENCY PRESERVATION POLICY

Every relational database undergoes decomposition during the process of normalization. It is therefore necessary for a decomposition to obey the lossless join property, because it guarantees that any decomposed relation can be recovered from its projections. Another important property of a decomposition of relation schema R into $\rho = (R_1, R_2 .... R_k)$ is that the set of dependencies F for R be implied by the projection of F onto the R_i's . We say that a decomposition ρ preserves a set of dependencies F, if the union of all the dependencies in $\rho_{R_i}(F)$, for i=1, 2......k logically implies all the dependencies in F. The necessity for ρ to preserve F is that the dependencies in F can be viewed as integrity constraints for the relation R and integrity constraints must never be relaxed/avoided otherwise the inconsistencies are bound to occur. If the projected dependencies do not imply F, then we should not represent R by $\rho = (R_1, R_2 .... R_k)$ otherwise we may find that the current value of R_i's represents a relation R that does not satisfy F, even if ρ had the lossless join property with respect to F. Alternatively every update to one of the R_i's would require a join to check that the constraints were not violated [22].

We therefore propose below a necessary condition for the dependencies to be preserved on joining two fuzzy relations using fuzzy equi join.

Theorem 1:- The dependency preservation property is obeyed in a type-2 fuzzy relational database only if the all the value of the joining attributes are atomic.

Proof) Let R and S be two type-2 fuzzy relations and F_1 and F_2 be two sets of fuzzy functional dependencies satisfied by R and S respectively. Let r_i & s_j be the joining attributes of the two fuzzy relations respectively. Let Q be the joined fuzzy relation i.e. $Q = R \ JOIN_F \ S$. Let r, s, q denote the instances of the R, S and Q respectively. Now, since a type-2 fuzzy relation schema may contain a crisp value (atomic) or a fuzzy set as a value of an attribute therefore depending on the domain of the attributes of R and S the following two cases exist.

Case I: All the values of the joining attributes are atomic (crisp)

Since a fuzzy set is non-atomic , this case is possible only when ,the domain of both the joining attributes are classical sets .This leads to case I in the fuzzy join operator[21].We will show that q satisfies $F_1 \cup F_2$ .Consider a ffd f: X→Y in $F_1 \cup F_2$. On the contrary assume that q does not satisfy f. Then by definition of ffd, there exists two tuples t_1 and t_2 in q such that $E_X[t_1, t_2] \neq 0$ and

$$\mu_{q[XY]}(t_1[X], t_1[Y]) \wedge \mu_{q[XY]}(t_2[X], t_2[Y]) \wedge E_X(t_1[X], t_2[X])$$
$$> E_Y(t_1[X], t_2[X]) \quad\quad (10)$$

Let t_k^r and t_k^s denote the projection of t_k over the attributes of r and s respectively, where k=1, 2.
Now from (10), $\mu_q(t_k) > 0$, k=1,2 and Using (3) and (9) , we have $\mu_q(t_k^r) > 0$ and $\mu_q(t_k^s) > 0$. Also

$$t_k^r[XY] = t_k[XY] \text{ and similarly } t_k^s[XY] = t_k[XY] \quad (11)$$

It can be noted that if any of the attributes X or Y or both is the joining attribute , then because of the atomicity of the values of the joining attributes , the binary equality values will hold (otherwise if none of the attributes X or Y is the joining attribute, the conclusion will be trivial) .

Now since $f \in F_1 \cup F_2$, thus f belongs to F_1 (or F_2) let us assume that $f \in F_1$, then both the attributes X and Y must belong to R. Considering the values of the two tuples t_1 and t_2 in r and using the equation (10), we derive a contradiction that the ffd f: X→Y does not hold in r. Hence q satisfies $F_1 \cup F_2$.

Case II: The values of the joining attributes are non atomic

This case exists when the join is executed by computing the fuzzy equality of two fuzzy sets. We will show that q may or may not satisfy the ffds in $F_1 \cup F_2$.

Consider a ffd f: X→Y in F_1, then obviously $f \in F_1 \cup F_2$ .We will show that q may not satisfy f.

Now since f is satisfied in r (as it belongs to F_1), thus by the definition of a fuzzy functional dependency, $\exists$ two tuples T_1 & T_2 in r with $T_1[X] = t_1, T_2[X] = t_2$ and $T_1[Y] = t_1'$ & $T_2[Y] = t_2'$ and $E_X(t_1, t_2) \neq 0$ such that the inequality

$$\mu_{r[XY]}(t_1, t_1') \wedge \mu_{r[XY]}(t_2, t_2') \wedge E_X(t_1, t_2) \leq E_Y(t_1', t_2') \quad (12)$$
holds.

Using LHS of the inequality (10) we have $\mu_r(T_1) \neq 0$ and $\mu_r(T_2) \neq 0$. Let us assume that the relation r is joined with relation s on the attributes Y and Z respectively where both Y and Z are defined over same universe of discourse. Let domain of Y is a set of fuzzy subsets with each fuzzy set containing more than one element but domain of Z is a classical set. Since the values of the joining attribute Y is non-atomic in nature (as it may contain a fuzzy set), it can be easily verified that due to incorporation of non-atomic values as the value of attribute Y, a fuzzy set occurring as a value of a Y in a tuple is fuzzily equal to more than one different values of Z (a crisp set). This is in contrast with the classical relational database. Hence in fuzzy relational databases, one tuple value of Y (a fuzzy set) can be joined with more than one value tuple values of Z , because more than one values of attribute Z in s can be partially equal to Y. The joined relation thus contains as many tuples as the number of values of Z which are partially equal to Y (note that in a classical relational database, one value of an attribute cannot be joined with distinct values of another attribute). Also, it is not

relevant to call them as spurious tuples because these multiple tuples represent the constraint actually holding on the fuzzy relations r and s.

Without loss of generality, let T_1^q & T_2^q be any two tuples in the relation q such that $\mu_q(T_1^q) > 0$ and $\mu_q(T_2^q) > 0$ and $E_X(T_1^q[X], T_2^q[X]) \neq 0$ but $E_Y(t_1'', t_2'') = 0$, where t_1'' and t_2'' denotes the values of Y in two tuples T_1^q & T_2^q. Now from construction, $E_Y(t_1'', t_2'') < E_X(T_1^q[X], T_2^q[X])$ always. Now $\mu_{q[X,Y]}(T_1^q[X], T_1^q[Y]) \neq 0$ & $\mu_{q[X,Y]}(T_2^q[X], T_2^q[Y]) \neq 0$ (by construction, otherwise these two tuples would not have been present in the relation q). Considering the inequality (12), we find that since $E_Y(t_1'', t_2'') = 0$, LHS $\nleq$ RHS, which shows that the ffd does not hold in q.

Thus we see that the fuzzy functional dependencies may not be preserved in joining two type-2 fuzzy relations. We next provide an example to this effect.

Example 3: Let R and S be two type-2 fuzzy relations given by R(A,B) and S(C,D) where domain of A is a classical set given by dom(A)= $\{a_1, a_2, a_3, a_4\}$ and domain of B is a set of fuzzy subsets defined over $\{b_1, b_2, b_3, b_4, b_5\}$. The fuzzy relation S has domain of C as an ordinary set with dom(C)= $\{b_1, b_2, b_3, b_4, b_5\}$ and D is a set of fuzzy subsets defined over $\{d_1, d_2, d_3\}$. The relational instance r and s of fuzzy relations R and S are given in Tables II (A) and II (B) respectively.

TABLE II (A)
TYPE-2 FUZZY RELATION R (A,B)

	A	B	μ
t_1	a_1	$\{0.1/b_1 + 0.7/b_2\}$	0.4
t_2	a_2	$\{0.3/b_1 + 0.4/b_2 + 0.5/b_3 + 0.6/b_4\}$	0.7
	a_3	$\{0.3/b_1 + 0.6/b_2\}$	0.5
	a_4	$\{0.1/b_1 + 0.3/b_3 + 0.7/b_5\}$	0.67

TABLE II (B)
TYPE-2 FUZZY RELATION S (C,D)

	A	B	μ
	a_1	$\{0.1/b_1 + 0.7/b_2\}$	0.4
	a_2	$\{0.3/b_1 + 0.4/b_2 + 0.5/b_3 + 0.6/b_4\}$	0.7

Let t_1 and t_2 be two tuples of R as shown in Table II (A). Let $t_1[B] = \{0.1/b_1 + 0.7/b_2\}$ is denoted by B_1 and $t_2[B] = \{0.3/b_1 + 0.6/b_2\}$ is denoted by B_2.

Then $I(B_1, B_2) = \{x \in domB : \mu_{B_1}(x) \leq \mu_{B_2}(x)\}$ which is calculated as

For $x = b_1$: $\mu_{B_1}(b_1) = .1$ and $\mu_{B_2}(b_1) = 0.3$, so $\mu_{B_1}(b_1) \leq \mu_{B_2}(b_1)$ for $x = b_1$

For $x = b_2$: $\mu_{B_1}(b_2) = 0.7$ and $\mu_{B_2}(b_2) = 0.6$, so $\mu_{B_1}(b_2) \nleq \mu_{B_2}(b_{21})$ for $x = b_2$

Similarly $\mu_{B_1}(b_3) \leq \mu_{B_2}(b_3)$ for $x = b_3$, $\mu_{B_1}(b_4) \leq \mu_{B_2}(b_4)$ for $x = b_4$ and $\mu_{B_1}(b_5) \leq \mu_{B_2}(b_5)$ for $x = b_5$. Hence, $I(B_1, B_2) = \{b_1, b_3, b_4, b_5\}$, similarly $I(B_2, B_1) = \{b_2, b_3, b_4, b_5\}$

Now $[\widetilde{\subseteq}(B_1, B_2)] = \underset{x \in domB}{\Lambda} \{(B_1^c \vee B_2)(x)\}$

$= \underset{x \in domB}{\Lambda} \{(1 - B_1) \vee B_2)(x)\} = \underset{x \in domB}{\Lambda} \{(1 - \mu_{B_1}(x)) \vee \mu_{B_2}(x)\}$

for $x = b_1$:
$\{(1 - \mu_{B_1}(b_1)) \vee \mu_{B_2}(b_1)\} = \{(1 - 0.1) \vee 0.3\} = (0.9 \vee 0.3) = 0.9$

for $x = b_2$:
$\{(1 - \mu_{B_1}(b_2)) \vee \mu_{B_2}(b_2)\} = \{(1 - 0.7) \vee 0.6\} = (0.3 \vee 0.6) = 0.6$

For $x = b_3$: $\{(1 - \mu_{B_1}(b_3)) \vee \mu_{B_2}(b_3)\} = \{(1 - 0) \vee 0\} = 1$

For $x = b_4$: $\{(1 - \mu_{B_1}(b_4)) \vee \mu_{B_2}(b_4)\} = \{(1 - 0) \vee 0\} = 1$

For $x = b_5$: $\{(1 - \mu_{B_1}(b_5)) \vee \mu_{B_2}(b_5)\} = \{(1 - 0) \vee 0\} = 1$

Therefore $[\widetilde{\subseteq}(b_1, b_2)] = 0.9 \wedge 0.6 \wedge 1 \wedge 1 \wedge 1 = 0.6$.

Also $[\widetilde{\subseteq}(B_2, B_1)] = \underset{x \in domB}{\Lambda} \{(1 - \mu_{B_2}(x)) \vee \mu_{B_1}(x)\}$

for $x = b_1$:
$\{(1 - \mu_{B_2}(b_1)) \vee \mu_{B_1}(b_1)\} = \{(1 - 0.3) \vee 0.1\} = (0.7 \vee 0.1) = 0.7$

for $x = b_2$:

$\{(1 - \mu_{B_2}(b_2)) \vee \mu_{B_1}(b_2)\} = \{(1 - 0.6) \vee 0.7\} = (0.4 \vee 0.7) = 0.7$

For $x = b_3$: $\{(1 - \mu_{B_2}(b_3)) \vee \mu_{B_1}(b_3)\} = \{(1 - 0) \vee 0\} = 1$

For $x = b_4$: $\{(1 - \mu_{B_2}(b_4)) \vee \mu_{B_1}(b_4)\} = \{(1 - 0) \vee 0\} = 1$

For $x = b_5$: $\{(1 - \mu_{B_2}(b_5)) \vee \mu_{B_1}(b_5)\} = \{(1 - 0) \vee 0\} = 1$

Therefore $[\widetilde{\subseteq}(B_2, B_1)] = 0.7 \wedge 0.7 \wedge 1 \wedge 1 \wedge 1 = 0.7$

Therefore $\underset{x \in domB}{\Lambda} \{\chi_{I(B_1, B_2)}(x)\} = 1 \wedge 0 \wedge 1 \wedge 1 \wedge 1 = 0$

Similarly $\underset{x \in domB}{\Lambda} \{\chi_{I(B_2, B_1)}(x)\} = 0 \wedge 1 \wedge 1 \wedge 1 \wedge 1 = 0$

Now by definition of $\chi_{I(b_1, b_2)}$,

$$\{\chi_{I(b_1, b_2)}(x)\} = \begin{cases} 1 & \text{if} \quad b_1 \subseteq b_2 \\ 0 & \text{if} \quad b_1 \not\subset b_2 \end{cases}$$

Hence
$[\widetilde{\subseteq}_*(B_1, B_2)] = [\widetilde{\subseteq}(B_1, B_2)] \vee (\underset{x \in domB}{\Lambda} \{\chi_{I(B_1, B_2)}(x)\}) = 0.6 \vee 0 = 0.6$

$[\widetilde{\subseteq}_*(B_2, B_1)] = [\widetilde{\subseteq}(B_2, B_1)] \vee (\underset{x \in domB}{\Lambda} \{\chi_{I(B_2, B_3)}(x)\}) = 0.7 \vee 0 = 0.7$

Therefore
$[\widetilde{\cong}_*(B_1, B_2)] = [\widetilde{\subseteq}_*(B_1, B_2)] \wedge [\widetilde{\subseteq}_*(B_2, B_1)] = 0.6 \wedge 0.7 = 0.7$

Using the above computations we find that the inequality (8) is satisfied for the two tuples t_1 and t_2 since

$$\mu_{r[AB]}(t_1[A], t_1[B]) \wedge \mu_{r[AB]}(t_2[A], t_2[B])$$
$$\wedge E_A(t_1[A], t_2[A]) \le E_B(t_1[B], t_2[B])$$
$$(= 0.4 \wedge 0.5 \wedge 1 \le 0.6)$$

TABLE III
AN INSTANCE OF FUZZY JOINED RELATION Q= R JOIN$_F$ S

A	B	D	μ
a_1	b_1	$\{0.1/d_1 + 0.3/d_2\}$	0.1
a_1	b_2	$\{0.5/d_2\}$	0.4
a_2	b_1	$\{0.1/d_1 + 0.3/d_2\}$	0.3
a_2	b_2	$\{0.5/d_2\}$	0.3
a_2	b_3	$\{0.3/d_1 + 0.7/d_3\}$	0.5
a_4	b_1	$\{0.1/d_1 + 0.3/d_2\}$	0.3
a_4	b_2	$\{0.5/d_2\}$	0.5
a_4	b_1	$\{0.1/d_1 + 0.3/d_2\}$	0.3
a_4	b_3	$\{0.3/d_1 + 0.7/d_3\}$	0.3

Similarly (8) can be verified for any two tuples of r. Thus we conclude that the ffd X → Y holds on r. We will now show that ffds may not be preserved on executing the fuzzy-join operator. Note that the attributes B of Rand C of S are defined over the same domain and therefore are fuzzy-join compatible. Let Q be the result of applying the fuzzy equi join operator on R and S. Let q denotes a relation instance of Q (Table III). Let F_1 and F_2 denote the set of ffds holding on R and S respectively. Since the ffd f: A⇢B holds in r of R, so f $\in F_1 \cup F_2$. We will prove that the f does not hold on q.

Now consider the two tuples t_1 = $(a_1, b_1, \{0.1/d_1 + 0.3/d_2\}, 0.1)$ and $t_2 = (a_1, b_2, \{0.5/d_2\}, 0.4)$ Here $\mu_{q[AB]}(t_1[A], t_1[B]) = 0.1$, $\mu_{q[AB]}(t_2[A], t_2[B]) = 0.4$ and $E_A(t_1[A], t_2[A]) = 1$ but $E_B(t_1[B], t_2[B]) = 0$ Hence the ffd A⇢B does not hold in q and consequently the ffd A⇢B does not hold in $F_1 \cup F_2$.

However consider another fuzzy relation schema S'(C', D) given below where the domain of C' is the set of fuzzy subsets defined over $\{b_1, b_2, b_3, b_4, b_5\}$ and D is a set of fuzzy subsets defined over $\{d_1, d_2, d_3\}$.

S'

C'	D	μ
$\{0.4/b_1\}$	$\{0.1/d_1 + 0.3/d_2\}$	0.4
$\{0.5/b_2\}$	$\{0.5/d_2\}$	0.5
$\{.8/b_3\}$	$\{0.3/d_1 + 0.7/d_3\}$	0.6

Let Q' denotes the fuzzy join of R with S' i.e Q' = R JOIN$_f$ S' and let q' be a relational instance of Q' given in (Table IV).

It can be easily verified that the ffd A⇢B holds in **Q'**. Thus we conclude that in order to preserve the fuzzy functional

dependencies in joining of two type-2 fuzzy relations, the fuzzy join must be executed over atomic values. Only the attributes which have their domain as ordinary sets can guarantee to provide the atomic values of the tuples.

VII. FUZZY JOIN DEPENDENCY

In this section we define the fuzzy join dependency in context of type-1 & type-2 fuzzy relational databases. The join dependency in a classical relational database verifies the lossless join decomposition of a relation schema. We try to achieve the same results in context of fuzzy relational databases.

Definition 5:- Given a type-1 or type-2 fuzzy relational schema R, Consider the following set of its projections [$R_1, R_2, \ldots, R_n$]. The relation schema R satisfies the fuzzy join dependency denoted as FJD[$R_1, R_2, \ldots, R_n$],if the only if the fuzzy equi join of the projection R_i on R, $1 \le i \le n$,is equal to R i.e.

$$R = P_{R_1}(R) \; JOIN_f \; P_{R_2}(R) \; JOIN_f \ldots P_{R_n}(R) \; JOIN_f$$

where $JOIN_f$ denotes the fuzzy equality operator

In the above definition, it can be noted that the fuzzy Join dependency holds iff the result of the fuzzy equi join of the decomposed fuzzy relations recovers to the original fuzzy relation.

TABLE IV
q' , A RELATIONAL INSTANCE OF Q' = R JOIN$_F$ S'

A	B	D	μ
a_1	$\{0.1/b_1\}$	$\{0.1/d_1 + .3/d_2\}$	0.1
a_1	$\{0.5/b_2\}$	$\{0.5/d_2\}$	0.4
a_2	$\{0.3/b_1\}$	$\{0.1/d_1 + .3/d_2\}$	0.3
a_2	$\{0.4/b_2\}$	$\{0.5/d_2\}$	0.4
a_2	$\{0.5/b_3\}$	$\{0.3/d_1 + .7/d_3\}$	0.5
a_1	$\{0.3/b_1\}$	$\{0.1/d_1 + .3/d_2\}$	0.3
a_1	$\{0.5/b_2\}$	$\{0.5/d_2\}$	0.5
a_4	$\{0.1/b_1\}$	$\{0.1/d_1 + 0.3/d_2\}$	0.1
a_4	$\{0.3/b_3\}$	$\{0.3/d_1 + 0.7/d_3\}$	0.3

Theorem 2: Let R be a type-1 fuzzy relation schema which is decomposed into n number of relation schemas $R_1, R_2 \ldots R_n$. If either $R_i = R$, for some i=1, 2….n or every R_i is a super-key of R then the fuzzy join dependency denoted by FJD($R_1, R_2 \ldots R_n$) holds on R.

Proof- Let r be a relational state of R. Since R is a type-1 fuzzy relation schema, it cannot have the domain of any attribute as set of fuzzy subsets. So, non-atomic values cannot appear as value of some attribute in a tuple. Let us discuss the two cases separately.

Case 1: Suppose $\exists$ **some** R_k **(k** $\in [1, n]$**) such that** $R_k = R$

Let $A_1, A_2.....A_n$ be the set of attributes such that $\bigcup_i A_i = R$ and $P_{R_i}(r) = A_i$, $\forall$ i = 1,2.....n.

According to the definition of fuzzy projection, the projection $r_i = P_{R_i}(r)$ is an s-ary fuzzy relation in the domain $(A_{i_1}) \times .... \times (A_{i_s})$. Also the membership function μ_{r_i} is given by

$$\mu_{r_i}(t) = \max_{t_r} \{\mu_r(t_r) \mid t_r[R_i] = t\}$$

Let t_1 and t_2 be the only two tuples of R such that $t_1[A_i] = t_2[A_i]$ (if more than two tuples exist, the proof can be provided in a similar way). Let us denote it by $t[A_i]$, then using the definition of projection,

$$\mu_{R_i}(t) = \vee (\mu_R(t_1), \mu_R(t_2)) \tag{13}$$

Now since $R_k = R$, joining of R_i with R_k will give rise to no additional tuple in $R_k' = R_k$ JOIN$_f$ R_i, because R_k will be joined with R_i on equality of $t_1[A_i]$ with $t[A_i]$ and $t_2[A_i]$ with $t[A_i]$. Now Since the fuzzy relation schema R was a type-1 fuzzy relation schema, both the values $t_1[A_i]$ (respectively $t_2[A_i]$) and $t[A_i]$ will be crisp values and the fuzzy equality value will be equal to 1 because of the derivation of $t[A_i]$ from $t_1[A_i]$ (respectively $t_2[A_i]$) and on every other tuple the value of fuzzy equality will be 0. The only point remains to be seen is the membership value of the tuple t_1 and t_2 in the fuzzily joined relation R_k'.

Now according to the definition of fuzzy equi-join, the membership values of the tuple t_1 in the fuzzily joined relation schema R_k' is given by

$$\mu_{R_k \underset{A_i \cong * A_i}{\text{JOIN}_f R_i}}(t) = \wedge (\mu_R(t_1), \vee ((\mu_R(t_1), \mu_R(t_2))))$$

$$= \mu_R(t_1)$$

Similarly the membership value of t_2 in R_k' is given by

$$\mu_{R_k \underset{A_i \cong * A_i}{\text{JOIN}_f R_i}}(t) = \wedge (\mu_R(t_2), \vee ((\mu_R(t_1), \mu_R(t_2))))$$

$$= \mu_R(t_2)$$

Hence, the membership values of the tuples t_1 and t_2 in the fuzzily joined relation R_k' come out to be the same value as previous one. Since R_i was any arbitrary chosen projection of R, the fuzzy join of $R_1, R_2.....R_n$ will give rise to R. Hence

$$\text{JOIN}_f (P_{R_1}(r), P_{R_2}(r).........P_{R_n}(r)) = r$$

And the fuzzy join dependency holds. In classical relation databases the join dependency has been referred as trivial join dependency when $R_i = R$ for some i.

Case 2: Every R_i is a super-key of R

Since every R_i is a super-key of R, it implies that every R_i contains key(it can be noted that R may contain more than one key of R viz. a primary key and several candidate keys, but we are talking in the notion that every R_i is superset of the same key. Using the property of a key, we observe that since every R_i is a super-key of R, unlike case1 no two tuples t_1 and t_2 of R can exist such that $t_1[A_i] = t_2[A_i] = t[A_i]$. Hence for any tuple T belonging to relation state r of R such that $T[A_i] = t[A_i]$ and we have $\mu_R(T[A_i]) = \mu_{R_i}(t[A_i])$. Rest of the argument can be given similar to as in case 1 and conclude that the FJD holds.

In the next theorem we derive the conditions for the fuzzy join dependency to hold in type-1 fuzzy relational schemas.

Theorem 3: Let r denotes a relational instance of a type-2 fuzzy relation R such that the key of R does not contain any attributes with domain as set of fuzzy subsets .Let $R_1, R_2.....R_n$ denote its fuzzy projections .If R_i is a super-key of R ,then the fuzzy join dependency (FJD): $\text{JOIN}_f (P_{R_1}(r), P_{R_2}(r).........P_{R_n}(r)) = r$ holds on R.

Proof: Since every R_i is a super-key of R, therefore R_i will be fuzzily joined with $R_j (i \neq j)$ on the key attribute. Since the key attributes will be having the crisp values as the values of the tuples, therefore the fuzzy join will be lossless. So that the fuzzy join dependency (FJD) $\text{JOIN}_f (P_{R_1}(r), P_{R_2}(r)$ $.........P_{R_n}(r)) = r$ holds (by case 2 of Theorem 2).

Again the atomicity plays a crucial role in joining two relations. Since the attributes with domain as a set of fuzzy subset can have non atomic tuple values, therefore joining two tuples on non-atomic values may give rise to a lossy join. The problem when the key attribute can have domain as a set of fuzzy subsets can be dealt only after removing the redundancy between two tuples (by fixing a minimum threshold value as a key always results in non-redundant tuples)and is not addressed in this article.

It can be noted that in case of a type-2 fuzzy relation schema the FJD is not guaranteed to hold even if the trivial join dependency is holding on the relation as projection R on $R_1, R_2.....R_n$ may not guarantee propagation of key in every R_i and since there may exist a case when two projection $R_i \& R_j$ $(i \neq j)$ are joined on an attribute having domain as set of fuzzy subsets (non-atomic values) and thus may result in a lossy join.

VIII. CONCLUSION

In this paper we have proposed a multi-relational data dependency called the fuzzy join dependency to verify the lossless join property of the decomposed relations. We observed that the subnormal/normal fuzzy sets may occur as value of an attribute in a type-2 fuzzy relational schema and the fuzzy equality of two subnormal/normal fuzzy sets may be zero/non-zero. We studied the conditions for such equality

over different domains. The fuzzy join dependency was defined on the basis of fuzzy equi-join operator. We conclude that atomicity of the attribute value is a necessary condition for the dependency preservation property of the ffds. In last we obtain the conditions for obtaining the lossless join of the decomposed schemas. We found that the non-atomic fuzzy sets such as "Low" which frequently occur as value of attributes are responsible for the lossy join. However, since in a type-2 fuzzy relational schema the occurrence of non-atomic sets is inevitable, therefore care must be taken while joining two schemas that they must not be joined over such attributes which contain values as non-atomic fuzzy sets.

IX. ACKNOWLEDGMENT

REFERENCES

[1] K.V.S.V.N. Raju, A.K. Majumdar, "The study of joins in fuzzy relational databases, Fuzzy sets and systems", 21(1987), 19-34.
[2] P. C. Saxena & B.K.Tyagi, "Fuzzy Functional Dependencies & independencies in extended fuzzy relational data models", 69(1995), 65-89.
[3] Bhattacharjee T.K and Mazumdar A.K, "Axiomatisation of fuzzy multivalued dependencies in a fuzzy relational data model, Fuzzy sets and systems", 96(1988), 343 –352.
[4] Jyoti S , Babu M. Syam, "Multivalued dependencies in fuzzy relational databases and lossless join decomposition, Fuzzy sets and systems", 88(1997),315-332.
[5] Liu Wei Yi, "Fuzzy data dependencies and implications of fuzzy data dependencies, Fuzzy Sets and systems", 92(1997), 341-348.
[6] P.C.Saxena & D.K.Tayal, "Multivalued dependencies in type-2 fuzzy relational data models using fuzzy functions", communicated to Iranian Journal of Fuzzy Systems.
[7] SÖzat Mustafa & Adnan Yazici, "A complete axiomatization for fuzzy functional dependencies and multivalued dependencies in fuzzy database relations, Fuzzy sets and systems", 117(2001), 161-181.
[8] Edward Sciore, "A complete axiomatisation of full join dependencies", Journal of ACM, 29(2) (1982), 323-393.
[9] K.P.Tan , "A less costly constraints checking for join dependency , In Proceedings of the Tenth International Conference on Very Large Data Bases", Singapore , August 1984.
[10] K.V.S.V.N.Raju , A.K.Majumdar, "Fuzzy functional dependencies and lossless join decomposition of fuzzy relational databases", ACM Transactions Database systems,13(2)(1988),129-166.
[11] F.E.Petry, "Fuzzy Databases Principles and Applications", Kluwer Academic publishers, USA 1996.
[12] Ma Z.M , Zhang W.J , Ma W.Y , Mili F, "Data Dependencies in extended Possibility based Fuzzy Relational Databases", International Journal of Intelligent Systems,17(2002),321-332.
[13] Chen G.Q ,Kerre E.E, Vandenbulcke J, "A computational Algorithm for FFD closure and a complete axiomatization of fuzzy functional dependency (FFD)", Int J Intelligent systems , 9(1994) ,421- 439.
[14] Chen G.Q, Kerre E.E., Vandenbulcke J, "Normalization based on fuzzy functional dependency in a fuzzy relational data model, J information systems", 21(3)(1996),299-310.
[15] Dubois Didier & Prade H, "Generalised Dependencies in fuzzy data bases", Journal of American Society for Information Science, 49(3),219-235.
[16] Prade Henri & Testamale C, "Generalising database relational algebra for the treatment of the incomplete or uncertain information and vague queries", Info Science, 34(1984),115-143.
[17] Umano M, "FREEDOM –O: A fuzzy database system, In fuzzy information and Decision Processes.", M M Gupta, E Sacnhez , Eds North Holland , Amsterdam 1982,339-347.
[18] Mustafa Demirci, "Fuzzy functions and their fundamental properties, Fuzzy sets and systems", 106(1999) ,239-246.
[19] M. Sasaki, "Fuzzy functions, Fuzzy sets and systems", 55(1993),295-301.
[20] P.C.Saxena & D.K.Tayal, "Normalisation in type-2 fuzzy relational data model based on fuzzy functional dependency using fuzzy functions", communicated to Journal of Intelligent and Fuzzy Systems
[21] P C Saxena & D K Tayal, 'Fuzzy Join operator in fuzzy relational databases", In Proc: National Conference on Energy, Communication and Computer Systems, February 2-4, Delhi, India
[22] Desai B C, "Introduction to Database Systems", Galgotia Publications, 1999.

P. C. Saxena was born in Delhi, India and acquired a degree of M.Sc. (Operational Research) and Ph.D. (Operational Research) from University of Delhi in 1970. He is a member of various professional bodies including IEEE. He is also a referee on the panel of various reputed national and international journals including the International Journal of Fuzzy Sets and Systems.

He is currently a professor and Ex-Dean in School of Computer and Systems Sciences, Jawaharlal Nehru University, India and worked earlier as a lecturer in University of Delhi, India. He has published more than 75 papers in International Journals of high repute.

Prof. Saxena is a member of various professional bodies like Computer Society of India, Institute of Electrical and Electronic Engineers, India etc.

Devendra K. Tayal was born in 1977 in Delhi, India. He has acquired the degrees of B.Sc.(H)(Maths), M.Sc.(Maths), M.Tech. (Computer Engg) & Ph.D. from Jawaharlal Nehru University, Delhi, India in 2006.

He is an Assistant Professor in Department of Computer Engg, Jaypee Institute of Information Technology University, Noida, UP, India. He also worked as a lecturer in University of Delhi for about three years.

Mr. Tayal has published about a dozen papers in Fuzzy Logic and Software Engg .His areas of interest include Theory of Computation, Algorithms, Software Engg., Database Management Systems, Data Mining and Fuzzy Logic.