Abstract—In this work, an extension of the database relational model which incorporates vague or imprecise data is presented. Specifically, we extend the concept of functional dependency to Fuzzy Attributes Tables. This extension is based on the use of a residuated lattice as a truthfulness value set. For this goal, the domains are enriched with fuzzy similarity relations, the atomic values of the tables become fuzzy, and the functional dependencies are also fuzzy and based on the similarity relations. Moreover, we introduce a sound and complete axiomatic system to manipulate these dependencies, named Simplification Logic for fuzzy functional dependencies.

Keywords: Fuzzy Logic, Fuzzy Functional Dependency, Fuzzy Relations.

I. INTRODUCTION

Nowadays there exists many papers that have established the advantages of having a Fuzzy extension of the relational model for databases [16]. Thus, we may affirm that there exists a consensus in the need to have a “good” extension of the classical Codd model to Fuzzy Logic. But it is not only a matter of logicians. Thus, there exists several database researchers that claim for this extension.

It is sufficient to cite [2] which is the document elaborated in a periodic meeting of database experts. In the paper they say: “Traditional DBMSs were applied to business data processing, which typically focused on numbers and character strings. …When one leaves business data processing, essentially all data is uncertain or imprecise”. The authors asked for a way to store imprecise data but also a way to express imprecise queries and get imprecise answers. In this paper we will deal only with the first part of the problem.

So we will present results related with the storage of fuzzy data, but we will also focus on the (fuzzy) relation among the data and their (fuzzy) integrity constraints, more concretely their Functional Dependencies (FD).

Contrary to the well accepted standardization of the classical relation model, there does not exist a widely accepted extension of the relational model to Fuzzy environments. Unfortunately, this lack of consensus have several structural obstacles which make difficult the agreement of the researches.

There exists some authors that have taken into account fuzzy values and so they consider fuzzy domains (a set of fuzzy values) and define a similarity relation which fuzzyfies the equality crisp relation between crisp attributes values [27]. Nevertheless, a significant part of these works do not extend tables to consider imprecise knowledge in them fand they consider the definition of a classical table over domains with similarity relations [7].

Fortunately, some authors have proposed a proper extension of the relational model to consider vagueness and uncertainty. As R. Belohlavek and V. Vychodil cited in [7]: “The link of fuzzy logic extensions of Codd’s model to a corresponding logical calculus (i.e., to a fuzzy logic in so-called narrow sense in this case) is missing or, at least, not handled appropriately”.

In our opinion this sentence holds the key of the success fuzzy extension of the relational model: the extension of domains is not good and we need to extend all the concepts of the relational model, building the new model on a solid fuzzy approach. The authors cited above have tackled this problem successfully, providing a proper fuzzy definition of table [7], an extension of the Relational Algebra [10], a proper definition of fuzzy functional dependencies [5], a complete inference rule for fuzzy FD [8], etc.

Functional Dependencies are the key elements where the role they play in the Relational Model is an appeal for several works which introduce different fuzzy functional dependencies. All of them affirm that they extend the classical definition in a proper Fuzzy FD.

Nevertheless some of these works preserves the original FD definition and substitutes the equality between values of an attribute by a similarity relation [18], [27], [30], [34]. A proper extension of the concept of functional dependencies demands that we are able to introduce the uncertainty in the FDs that are holds by a relation by associating a grade to each FDs [29].

W. Armstrong [1] introduces a set of inference rules, named Armstrong Axioms, which lead to the first complete axiomatic system for FDs. The goal of the Armstrong’s work was to illustrates the semantics of the FDs. Thus, his system and other very similar subsequent systems [3], [12], [13] are strongly based in the transitivity, which limits the design of a FD automated deduction method. There exists several authors that have extended the Armstrong axioms to provide a complete inference system for fuzzy functional dependencies [6], [29], [30], [34]. There exists several differences among these works, because they manage different Fuzzy FDs definition, but all of them lack executable skill to produce automated methods. These methods may be consider as fuzzy extensions of the Armstrong’s Systems and thus, they are also based on a (fuzzy) transitivity rule.

In [18] we introduce a definition of fuzzy functional dependencies over domains with similarity relations. We have develop a new kind of inference system for classical FDs
which avoids the use of the transitive rule and opens the door for the design of automated deduction methods for FDs. This logic, named Simplification Logic for Functional Dependencies, have provide a framework to develop some execution methods to deal with some classical FDs problems [17], [25], [26]. Our fuzzy FD notion was introduced to have a proper fuzzy definition which allows us to built a Fuzzy extension of the SLFD Logic [19], [20].

Our first fuzzy extension of the relational model includes a proper definition of fuzzy FD in the sense we mention above and it allows us to build a fuzzy extension of the SLFD logic in a natural way. Nevertheless, it lacks a proper fuzzy notion of relational table, as those introduced in [5], [7] where tuples are labelled with an uncertainty grade. In this work we introduce a new fuzzy relational model where uncertainty is included in the atoms, i.e., each attribute of each tuple in the relation has a different fuzzy value. This new concept, named Fuzzy Attributes Tables, allows us to introduce a more general fuzzy relational model than those proposed in [5], [7]. Besides that, a fuzzy extension of the SLFD Logic like those proposed in [19], [20] is presented.

II. BACKGROUND AND PRELIMINARIES

In this section, we show the basic concepts of the Relational Model, with emphasis on functional dependencies, and we also outline the basic concepts of Fuzzy Logic. The section ends with a description of the notion of Fuzzy Functional Dependency.

A. Classical Database preliminaries

Let \( \{D_a \mid a \in A\} \) be a family of sets indexed in a finite non-empty set of indexes \( A \). We call attributes to the indexes and domain of the attribute \( a \) to the set \( D_a \). We work over the product of these domains, \( \mathbb{D} = \prod_{a \in A} D_a \). The elements in this product \( t = (t_a)_{a \in A} \in \mathbb{D} \) will be named tuples. A relation is a set of tuples \( R \subseteq \mathbb{D} \), usually represented as a table.

We introduce here the notation widely accepted in the database community. Given \( X, Y \subseteq A \), \( XY^+ \) denotes \( X \cup Y \). Given \( X \subseteq A \), \( D_X \) denotes \( \prod_{a \in X} D_a \). Let \( t \in R \) be a tuple, then \( t/_{X} \) denotes the projection of \( t \) to \( D_X \); that is, if \( t = (t_a)_{a \in A} \) then \( t/_{X} = (t_a)_{a \in X} \).

**Definition 2.1:** Any affirmation of the type \( X \rightarrow Y \), where \( X, Y \subseteq A \), is named a functional dependency. We say that a relation \( R \subseteq \mathbb{D} \) satisfies \( X \rightarrow Y \) if, for all \( t_1, t_2 \in R \) we have that: \( t_1/_{X} = t_2/_{X} \) implies that \( t_1/_{Y} = t_2/_{Y} \).

**Remark 2.2:** The term functional comes from the fact that: the relation \( R \) satisfies the FD \( X \rightarrow Y \) if \( R \) restricted to \( XY \) is a (partial) function from \( D_X \) to \( D_Y \).

B. Fuzzy Preliminaries

Unlike Codd’s model is named classical relational model, it is not accurately crisp. A certain degree of vagueness is introduced because in all domains the null value is included [29]. Besides that, the Relational Algebra and the Relational Calculus deal with null values in a consistent and correct manner. Thus, the Codd’s model provides a way to specify and manage uncertainty. Nevertheless it is based on a three-valued logic and it may not be consider a proper fuzzy approach.

To fuzzify this concept it is necessary to replace the truth values sets by other algebraic structure, which allows us to represent truth degrees. The easiest approach is to consider the \( ([0, 1], \min, \max, 0, 1) \) lattice. Moreover, it is usual to enrich this lattice with a t-norm and with an residuated implication defined over the t-norm. This is only a particular case of a complete and residuated lattice \(^1\).

Then, our approach uses the unit interval \([0, 1]\) for the system of truth values, the infimum (denoted by \( \wedge \)) as the universal quantifier, the supremum (denoted by \( \vee \)) as the existential quantifier, an arbitrary t-norm (triangular norm denoted by \( \otimes \)) as the conjunction and the residue defined from this t-norms \( a \rightarrow b = \sup \{ x \in [0, 1] \mid x \otimes a \leq b \} \). That is, the system of truth values is the residuated complete lattice \([0, 1], \vee, \wedge, 0, 1, \otimes, \rightarrow \) where \([0, 1], \otimes, 1\) is a commutative monoid, \( \otimes \) is isotonic \( a \leq b \) implies \( a \otimes c \leq b \otimes c \), for all \( a, b, c \in [0, 1] \) and \( (\otimes, \rightarrow) \) is an adjoin pair \( (a \otimes b \leq c \) if and only if \( a \leq b \rightarrow c \).

The following properties, satisfied in every residuated lattice, are well-known:

\[
\begin{align*}
RL_1) & a \rightarrow (b \wedge c) = (a \rightarrow b) \wedge (a \rightarrow c) \\
RL_2) & (a \wedge b) \rightarrow c = (a \rightarrow c) \vee (b \rightarrow c) \\
RL_3) & a \otimes (a \rightarrow b) \leq b \\
RL_4) & a \otimes (b \wedge c) \leq (a \otimes b) \wedge (a \otimes c)
\end{align*}
\]

C. Fuzzy Functional Dependencies

The most usual way to fuzzify the concept of functional dependency is by replacing the equality in the definition by fuzzy relations named similarity relations.

In this case, we will consider that each domain \( D_a \) is endowed with a similarity relation \( \rho_a : D_a \times D_a \rightarrow [0, 1] \), that is, a reflexive \( (\rho_a(x, x) = 1 \) for all \( x \in D_a \)) and symmetric \( \rho_a(x, y) = \rho_a(y, x) \) for all \( x, y \in D_a \) fuzzy relation. Given \( X \subseteq A \), we can extend these relations to the set \( \mathbb{D} \) as follows: for all \( t, t' \in \mathbb{D} \),

\[
\rho_X(t, t') = \bigwedge_{a \in X} \rho_a(t_a, t'_a)
\]

Particularly, \( \rho_a(t, t') = 1 \)

**Remark 2.3:** The definitions of some fuzzy functional dependency in the literature [27], [30], [34] are very similar, having slight differences among them. They fuzzify the equality between the attributes value in the following way: A relation \( R \subseteq \mathbb{D} \) satisfies the fuzzy functional dependency \( X \rightarrow Y \) if \( \rho_X(t, t') \leq \rho_Y(t', t') \) holds, for all \( t, t' \in R \).

Although the definition introduces the fuzzy relations of similarity and generalizes the classical definition, we can say that the functional dependency remains crisp. The inclusion of multi-lattices [24], and even generalizations of the lattice structure as multi-lattices [23] are used.

\(^1\)Residuated lattices are used in the framework of Formal Concept Analysis, where \( X \rightarrow Y \) determine an attribute implications. Sometimes, multi-adjoint lattices [24], and even generalizations of the lattice structure as multi-lattices [23] are used.
of a degree of fuzzyness in the dependency itself is done in [29]. In [20] we generalize this definition of fuzzy functional dependency as follows.

**Remark 2.4:** A fuzzy functional dependency is an expression $X \rightarrow^{\theta} Y$ where $X, Y \subseteq A$ and $\theta \in [0,1]$. A relation $R \subseteq D$ is said to satisfy $X \rightarrow^{\theta} Y$ if, for all $t, t' \in R$, the inequation $\theta \otimes \rho_X(t, t') \leq \rho_Y(t, t')$ holds.

Given the adjointness property, this definition is equivalent to consider that

$$\theta \leq \bigwedge_{t,t' \in R} \rho_X(t, t') \rightarrow \rho_Y(t, t') \quad (1)$$

Obviously, if $\theta = 1$, the previous definition of FFD matches up with the definition of FFD proposed in Remark 2.3. Moreover, if the similarity relations are strongly reflexives ($\rho_a(x, y) = 1$ if and only if $x = y$) then it define a crisp classical functional dependency.

In this previous work [20] we also give a sound and complete axiomatic system for this semantic definition of fuzzy functional dependency. Section V will be focussed on inference system for fuzzy functional dependencies.

### III. A Fuzzy Extension of the Relational Model

With the Relational Algebra, Codd provides two relational operators (projection and selection) which allows us to query for the atomic elements of the Relational Model: a certain value for a given attribute of a table. The classical atoms, as we have mentioned in Section II-B, may received a value of the corresponding domain or a null value.

Most database fuzzy extensions are based on the establishment of fuzzy similarity relations between values of the domains. But, the table definition remains classical [27], [30], [34]. However, in [7], [8], [9] the authors introduce the notion of ranked data tables over domains with similarity. In their model, each tuple of the table is associated with a value in $[0,1]$ (the ranking) which represents the precise degree to which the tuple belongs to the relation (see Table II).

We consider that this model may be further generalized, so that a different ranking was associated to each attribute of the tuple. Since the value of each attribute, as we have mention, was conceived as the atomic element of the Relational Model, this approach provides the maximum level of uncertainty in fuzzy tables.

#### A. Fuzzy Attributes Tables

Thus, we name this extension of the classical relational table as Fuzzy Attributes Tables. We introduce a generalization of previous fuzzy data tables that will be maps

$$R: D \rightarrow [0,1]^A$$

That is, for each tuple $t = (t_a)_{a \in A} \in D$, the map $R$ gives a tuple of truth values $R(t) = (r_a)_{a \in A}$. So, for all $a \in A$, $r_a$ is the truthfulness of the value $t_a$ in the tuple $t$.

For each tuple $t$, $t_a$ denotes the value of the attribute $a$ in the tuple $t$ and $R(t)(a)$ is the truthfulness of the value $t_a$. Note that, it is possible that $R(t)(a) = 0$ for all attribute $a \in A$.

When we work with Fuzzy Attributes Tables, we also consider similarity relations in domains in the same way as previous works [18], [19], [20]. Fuzzy Attributes Table is an extension of the original table in the classical relational model by adding the degree of certainty to the values of each attribute.

#### Example 3.1:

First, we consider the set of attributes $A = \{n, h, s, g, y, f\}$ where $n$ represent the name, $h$ the hair, $s$ the skin, $g$ the age, $e$ the colour of the eyes and $f$ the sunscreen factor of the person.

Let be $D_n = \{\text{John, Albert, Mary, etc.}\}$, $D_h = \{\text{black, brown, blond, auburn, chestnut, red, gray, white}\}$, $D_s = \{\text{very-light, light, light-intermediate, dark-intermediate, dark, very-dark}\}$, $D_g = \{n \in \mathbb{N} \mid n \geq \text{min}_g, n \leq \text{max}_g\}$, $D_y = \{\text{Amber, Blue, Brown, Gray, Green, Hazel, Red}\}$ and $D_f = \{n \mid n \geq \text{min}_f, n \leq \text{max}_f\}$ the domains of the attributes.

We build the similarity relations in each domain $D_a$ as follows:

$$\rho_n(t_n, t'_n) = \begin{cases} 1 & \text{if } t_n = t'_n \\ 0 & \text{if } t_n \neq t'_n \end{cases}$$

$$\rho_y(t_y, t'_y) = \begin{cases} 1 & \text{if } t_y = t'_y \\ 0 & \text{if } t_y \neq t'_y \end{cases}$$

$$\rho_h(b, o) = 0.1, \rho_h(o, a) = 0.9, \rho_h(a, c) = 0.7, \rho_h(c, r) = 0.3, \rho_h(r, g) = 0.1$$

$$\rho_s(\text{v}, \text{l}) = 0.1, \rho_s(\text{l}, \text{i}) = 0.8, \rho_s(\text{i}, \text{a}) = 0.6, \rho_s(\text{a}, \text{d}) = 0.4, \rho_s(\text{d}, \text{g}) = 0.1$$

$$\rho_f(t_f, t'_f) = 1 - \frac{|t_f - t'_f|}{\text{max}_f - \text{min}_f}$$

And finally, we consider the following Fuzzy Attributes Table.

The information represented in the table corresponds to the information about persons and their characteristics. Note that

2 The values $\text{max}_g$, $\text{min}_g$, $\text{max}_f$ and $\text{min}_f$ are the maximum and minimum age and sunscreen factor respectively.
and, when \( \theta = 1 \), it matches up with the definition of FFD proposed in Remark 2.3. Moreover, if the similarity relations are strongly reflexives then it define a crisp classical functional dependency.

**Example 3.3:** In the Fuzzy Attributes Table given in Example 3.1 the fuzzy functional dependency \( \text{skin, eyes} \xrightarrow{\theta} \text{factor} \) holds for \( \theta = 0.7 \).

For instance, for the two first tuples, we have

\[
\rho^0_{by}(t_1, t_2) \rightarrow \rho^1_{fy}(t_1, t_2) = 0.6 \rightarrow 0.3 = 0.7
\]

This is the minimum for every pair of tuples in the table. So, fuzzy attribute table satisfies \( \text{skin, eyes} \xrightarrow{\theta} \text{factor} \) for all \( \theta \leq 0.7 \).

### IV. COMPARISON WITH OTHER FUZZY EXTENSIONS

As K. Raju and A. Majumdar cited in [27], fuzzy extension of the relations may be classify in two groups: in the type-1 approaches the fuzziness is only captured in the association among entities.

Thus, domains may be a classical set or a fuzzy set with its corresponding similarity relation or possibility distribution. In these approaches, the table definition remains classical. Type-2 provides further generalizations by allowing domains to be even a set of fuzzy sets (or possibility distributions). As the authors cited, such relations can be considered a second-level generalization of classical relations.

The extensions that are classified in the type-1 preserves the classical table definition and do not extend it. The second group of works are closer to the authors who claim for the inclusion of multi-valued attributes [32].

There exists other authors that consider as a basis for their fuzzy extensions some relational models which are not normalized into first normal form [29]. We consider that 1FN normalization is a basic requirement for a fuzzy extension of relational model, and therefore we use 1FN normalized tables and single-valued attributes. However, only considering \( D^a \) instead of \( D_a \) and defining the similarity relation \( \rho_a(M, N) = \bigwedge_{m \in M} \bigvee_{n \in N} \rho_a(m, n) \), our model is generalized to no normalized models.

As A. Ma and L. Yan establish [22], there exists a set of works which allows the association of fuzzy values to the tuples of the relations [27], [31]. In our opinion, there exists two authors that has successfully follow this line: R. Belohavek and R. Vychodil [7], [8], [9]. They have introduced **Ranked Data Tables** as an extension of the classical relational model. Ranked data tables over domains with similarity relations are given by domains, similarities and ranking. The ranking is the degree of each tuple \( t \) given by \( D(t) \). We illustrate this approach with the following example:

**Example 4.1:** With the same domain of the Example 3.1, the next table is a ranked data table. The extra column represents the ranking of each tuple.

The ranking represent the degree of certainty associated to each tuple based on our knowledge of the system. The authors
satisfies a fuzzy FD $X \Rightarrow Y$ as follows:

$$||X \Rightarrow Y||_D = \int_{t, t'} (t \approx_X t' \rightarrow t \approx_Y t')$$

where

$$t \approx_X t' = D(t) \otimes D(t') \rightarrow \bigwedge_{a \in X} \rho_a(t, t')$$

Note that, we have rewrite the definition given in [7] considering that the sets $X$ and $Y$ are crisp.

By other size, Ranked Data Tables may be considered a particular case of our Fuzzy Attributes Tables. The particularization may be done at least in two ways:

1. The Fuzzy Attributes Table, $R^\circ$, is obtained by copying the columns of the table, by assigning the truthfulness 1 in all value of this table and by introducing a new column, named ranking, where the rankings of each tuple are stored. Thus, the Fuzzy Attributes Table given in Example 4.1 (Table II) renders the following Table:

![Table II](image)

<table>
<thead>
<tr>
<th>name</th>
<th>hair</th>
<th>skin</th>
<th>age</th>
<th>eyes</th>
<th>factor</th>
<th>ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Black</td>
<td>dark</td>
<td>34</td>
<td>Brown</td>
<td>10</td>
<td>Yes/0.8</td>
</tr>
<tr>
<td>Albert</td>
<td>Brown</td>
<td>light</td>
<td>32</td>
<td>Blue</td>
<td>50</td>
<td>Yes/0.6</td>
</tr>
<tr>
<td>Mary</td>
<td>Auburn</td>
<td>light</td>
<td>26</td>
<td>Blue</td>
<td>50</td>
<td>Yes/0.6</td>
</tr>
<tr>
<td>Dave</td>
<td>Red</td>
<td>light</td>
<td>29</td>
<td>Blue</td>
<td>50</td>
<td>Yes/0.4</td>
</tr>
<tr>
<td>Noa</td>
<td>White</td>
<td>dark</td>
<td>32</td>
<td>Green</td>
<td>30</td>
<td>Yes/0.1</td>
</tr>
</tbody>
</table>

![Table III](image)

$R^\circ$ obtained from Table II

2. The Fuzzy Attributes Table, $R_\rho$, is obtained by defining $R(t)(a) = D(t)$ for all $a \in A$. For example, from the ranking data table (Table II) in Example 4.1, the fuzzy attribute table (Table IV) is obtained.

![Table IV](image)

$R_\rho$ obtained from Table II

In both cases, the initial ranking can be recovered by considering $D(t) = \bigwedge_{a \in A} R(t)(a)$. We have decided to use the second approach which does not change the original table schema. With this second approach the definition of fuzzy functional dependency for Fuzzy Attributes Tables presented in Section III-B coincides with the definition given in [7], as the following equality shows:

$$||X \Rightarrow Y||_D = \sup_{\theta | R_\rho \text{ satisfies } X \rightarrow_Y Y}$$

where $R_\rho$ is the fuzzy attribute table obtained from the ranked data table. We remark that, in this case, $\approx_X$ and $\rho_X^{R_\rho}$ are the same relation.
As a general conclusion from this comparison, there exists two dimensions to observe the fuzziness degree of an extended relational model. As we have illustrated, our approach reaches the maximum degree in the first dimension (the extension of the table definition) while Belohvake & Vichodyl get the high level in the expressiveness of the functional dependencies.

To the best of our knowledge, there does not exists a model which covers both dimensions in the top level. In the following table we resume the conclusion of this comparison:

<table>
<thead>
<tr>
<th></th>
<th>Classical table (fuzziness in domains)</th>
<th>Fuzziness in tuple</th>
<th>Fuzziness in attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classical FD</td>
<td>[15]</td>
<td>[27]</td>
<td>[27]</td>
</tr>
<tr>
<td>FD with a degree of fuzziness</td>
<td>[19]</td>
<td>[34]</td>
<td>[34]</td>
</tr>
<tr>
<td>FD with different degree in both sides</td>
<td>[28], [33]</td>
<td>[7]</td>
<td>[7]</td>
</tr>
</tbody>
</table>

V. THE SIMPLIFICATION LOGIC FOR FFDs

We are interested in an axiomatic system that allows us to syntactically derive FFDs. There exists in the literature some complete axiomatic system defined over FFDs with similarity relations [34], [30], [6].

In [20] we introduce the \( S_{FFD} \) logic, a logic for fuzzy functional dependencies which substitutes the transitivity rule by a novel simplification rule with an executable vocation. This logic is extended in the following subsection for our Fuzzy Attributes Tables and their functional dependencies.

A. \( S_{FFD} \) Logic

We introduce \( S_{FFD} \), a logic more adequate for the applications, named Simplification Logic for fuzzy functional dependencies. Its language is the following:

**Definition 5.1:** Given a numerable set of attribute symbols \( A \), we define the language

\[
L = \{ X^\theta \rightarrow Y \mid \theta \in [0, 1] \} \quad \text{and} \quad X, Y \in 2^A \quad \text{with} \quad X \neq \emptyset \}
\]

The semantic of this logic was outlined in previous sections. The semantic models are pairs made up of a family of domains with its similarities \( \{ (D_a, \rho_a) \mid a \in A \} \) and a fuzzy data table \( R : D \rightarrow [0, 1]^A \). However, to simplify the notation, we will only refer to the fuzzy data table.

So \( R \models X^\theta \rightarrow Y \) denotes that \( R \) satisfies the functional dependency \( X^\theta \rightarrow Y \), \( R \models \Gamma \) denotes that \( R \) satisfies every fuzzy functional dependency in the set \( \Gamma \) and \( \Gamma \models X^\theta \rightarrow Y \) denotes that \( R \models \Gamma \) implies \( R \models X^\theta \rightarrow Y \).

In this point we present the axiomatic system:

**Definition 5.2:** The axiomatic system \( S_{FFD} \) on L has one axiom scheme:

Reflexive Axioms (Ax): for all \( Y \subseteq X \)

\( \models_{S_{FFD}} X^1 \rightarrow Y \)

The inferences rules are the following:

Inclusion Rule (InR): if \( \theta_1 \geq \theta_2 \)

\( X^{\theta_1} \rightarrow Y \models_{S_{FFD}} X^{\theta_2} \rightarrow Y \)

Decomposition Rule (DeR): if \( Y' \subseteq Y \)

\( X^{\theta} \rightarrow Y \models_{S_{FFD}} X^{\theta} \rightarrow Y' \)

Composition Rule (CoR):

\( X^{\theta_1} \rightarrow Y, \quad U^{\theta_2} \rightarrow V \models_{S_{FFD}} X^{\theta_1 \wedge \theta_2} \rightarrow YV \)

Simplification Rule (SiR): if \( X \subseteq U \) and \( X \cap Y = \emptyset \)

\( X^{\theta_1} \rightarrow Y, \quad U^{\theta_2} \rightarrow V \models_{S_{FFD}} X^{\theta_1 \wedge \theta_2} \rightarrow VY \)

The deduction (\( \models_{S_{FFD}} \)) and equivalence (\( \equiv_{S_{FFD}} \)) concepts are introduced as usual.

**Definition 5.3:** Let \( \Gamma, \Gamma' \subseteq L \) and \( \phi \in L \). We say that \( \phi \) is deduced from \( \Gamma \) in \( S_{FFD} \), denoted \( \Gamma \models_{S_{FFD}} \phi \), if there exists \( \varphi_1, \ldots, \varphi_n \in L \) such that \( \varphi_n = \phi \) and, for all \( 1 \leq i \leq n \), we have that \( \varphi_i \in \Gamma \). \( \varphi_i \) is an axiom or is obtained by applying the inference rules in \( S_{FFD} \) to formulas in \( \{ \varphi_j \mid 1 \leq j < i \} \).

We say that \( \Gamma \) and \( \Gamma' \) are *equivalent*, denoted \( \Gamma \equiv_{S_{FFD}} \Gamma' \), if \( \Gamma \models_{S_{FFD}} \phi \), for all \( \phi \in \Gamma \), and \( \Gamma' \models_{S_{FFD}} \phi \), for all \( \phi \in \Gamma \).

B. Soundness of \( S_{FFD} \) Logic

**Theorem 5.4:** The \( S_{FFD} \) axiomatic system is sound.

Proof: The soundness of Ax is due to, for all \( t, t' \in \mathbb{D} \), if \( Y \subseteq X \) then \( \rho_X^A(t, t') \rightarrow \rho_Y^A(t, t') = 1 \) because \( \rho_X^A(t, t') \leq \rho_Y^A(t, t') \).

The soundness of InR is trivial and DeR is sound because \( \rightarrow \) is isotone in the right side. The composition rule, CoR, is sound because, for all \( t, t' \in \mathbb{D} \),

\[
\rho_X^A(t, t') \rightarrow \rho_Y^A(t, t') \models_{S_{FFD}} \rho_Y^A(t, t') \rightarrow \rho_Y^A(t, t') \models_{S_{FFD}} \rho_Y^A(t, t') \rightarrow \rho_Y^A(t, t') \models_{S_{FFD}} \rho_Y^A(t, t') \rightarrow \rho_Y^A(t, t')
\]

where in (1) we have used that \( \rightarrow \) is antitone in the first argument, and, in (2), we have used that \( \rightarrow \) is right-distributive with respect to \( \wedge \) (that is, Property RL).
where (1) is due to \( a \geq a \otimes b, U \cap Y \subseteq Y \). Property RL3 and the antitone of \( \rightarrow \) on the first argument, (2) is a consequence of RL3 and (3) is true because \( X \subseteq U \cap Y \).

Finally, by using the adjointness property and the isomoritc of \( \rightarrow \) on the second argument, the following result is obtained

\[
(p_{X}^{+}(t, t') \rightarrow p_{X}^{+}(t, t')) \otimes (p_{X}^{+}(t, t') \rightarrow p_{X}^{+}(t, t')) \\
\leq p_{X,Y}^{+}(t, t') \rightarrow p_{X,Y}^{+}(t, t')
\]

\[\blacksquare\]

C. Completeness of SL._{FFD} Logic

The implication between the syntactic and the semantic level requires several previous definitions.

**Definition 5.5:** Let \( \Gamma \) be a set of fuzzy functional dependencies over \( A \). The closure of \( \Gamma \) is the set \( \Gamma^+ = \{ X \theta \rightarrow Y \mid \Gamma \vdash_{SL_{FFD}} X \theta \rightarrow Y \} \).

Note that, as a consequence of Ax and InR, \( \Gamma^+ \) assigns an infinite set of pairs \( (Y, \theta) \) to every non-empty set \( X \). If the set \( Y \) is also fixed then \( \Gamma^+ \) gives an interval (consequence of InR) whose supremum will be denoted as \( \theta^+_{X,Y} \)

\[
\theta^+_{X,Y} = \sup\{ \theta \in [0, 1] \mid X \theta \rightarrow Y \in \Gamma^+ \}
\]

On the other hand, if we fix the value of \( \theta \) then a subset of \( 2^A \) is obtained. This set is finite and, by DeR and CoR, is an ideal of \( (2^A, \subseteq) \). The maximum element of this ideal will be denoted by \( X^+_{\theta} \).

\[
X^+_{\theta} = \max\{ Y \subseteq A \mid X \theta \rightarrow Y \in \Gamma^+ \}
\]

The following theorem ensures the soundness and completeness of the axiomatic system.

**Theorem 5.6:** The \( S_{FFD} \) axiomatic system is complete.

**Proof:** The completeness is proved showing that \( \Gamma \vdash_{S_{FFD}} X \theta \rightarrow Y \) implies that \( \Gamma \mid \not\vdash X \theta \rightarrow Y \). That is, if \( Y \not\subseteq X^+_{\theta} \) then there exists a model for \( \Gamma \) that it is not model for \( X \theta \rightarrow Y \).

Let \( \Theta = \{ \theta^+_{U,V} \mid U, V \in 2^A, U \neq \emptyset \text{ and } \theta^+_{U,V} < \theta \} \) and let \( \tau \in [0, 1] \) such that \( \theta > \tau > \max \Theta \).

For all \( a \in A \), let \( D_{a} = \{ u, v \} \) and the similarity relation \( \rho_{a} \) defined in \( D_{a} \) such that

- \( \rho_{a}(u, u) = \rho_{a}(v, v) = 1 \) and
- \( \rho_{a}(u, v) = \rho_{a}(v, u) = \tau \).

Let also \( t, t' \in D \) with

- \( t_{a} = u \), for all \( a \in A \),
- \( t'_{a} = v \), for all \( a \in A \).

We will prove that the fuzzy data table \( R \) where

- \( R(t)(a) = R(t')(a) = 1 \) for all \( a \in A \) and
- \( R(t^{\ell})(a) = 0 \) for all \( t^{\ell} \in D \) with \( t^{\ell} \neq t \) and \( t^{\ell} \neq t' \) and for all \( a \in A \),

is a model for \( \Gamma \) but it is not model for \( X \theta \rightarrow Y \).

Note that, in the following, we will only need to study the similarity of tuples \( t \) and \( t' \) because, in other cases, \( \rho_{X}^{+}(t^{\ell}, t'^{\ell}) \rightarrow \rho_{X}^{+}(t^{\ell}, t'^{\ell}) = 1 \), for all \( \emptyset \neq U \subseteq A \) and \( V \subseteq A \).

Let \( U \theta_{1} \rightarrow V \in \Gamma \). If \( \theta^+_{U,V} < \theta \) then

\[
\rho_{X}^{+}(t, t') \rightarrow \rho_{X}^{+}(t, t') \geq 1 \rightarrow \tau = \tau > \theta^+_{U,V} \geq \theta_{1}
\]

If \( \theta^+_{U,V} \geq \theta \) and \( U \not\subseteq X^+_{\theta} \) then \( \nu_{U}(t_{1}, t_{2}) = \tau \) and

\[
\rho_{X}^{+}(t, t') \rightarrow \rho_{X}^{+}(t, t') \geq \tau \rightarrow \tau = 1 > \theta_{1}
\]

If \( \theta^+_{U,V} \geq \theta \) and \( U \subseteq X^+_{\theta} \) then \( V \subseteq X^+_{\theta} \) (it is a consequence of the composition rule) and

\[
\rho_{X}^{+}(t, t') \rightarrow \rho_{X}^{+}(t, t') = 1 \rightarrow \tau = \tau < \theta
\]

Therefore, \( R \) is a model for \( \Gamma \) but it is not a model for \( X \theta \rightarrow Y \) because

\[
\rho_{X}^{+}(t, t') \rightarrow \rho_{X}^{+}(t, t') = 1 \rightarrow \tau = \tau < \theta
\]

The previous theorems ensure the soundness and completeness of \( S_{FFD} \) axiomatic system. That is,

\( \Gamma \vdash_{S_{FFD}} X \theta \rightarrow Y \) if and only if \( \Gamma \models X \theta \rightarrow Y \)

VI. Conclusions and Future Works

In this paper a fuzzy database relational model is proposed. The development has a solid theoretical framework: residuated lattices as truthfulness value set. We have considered fuzziness in the (atomic) attributes values of the tables and a appropriate fuzzy similarity relations has been defined. Definitions for Fuzzy Attributes Tables and fuzzy functional dependencies are provided. Besides that, the logic Simplification Logic for fuzzy functional dependencies (SL._{FFD}) is presented in this model.

The model here presented can be translated into Formal Concept Analysis. The use of the SL._{FFD} logic to manipulate attribute implications in Formal Concept Analysis is begin developed.

Moreover, the extension of our fuzzy model for considering fuzziness on the left and right hand sides of the fuzzy functional dependencies is a short-term future work. The extension to no first normal form models will be tackled in the future.

Finally, the interesting possibilities of SL._{FFD} logic for the development of novel methods to manipulate fuzzy functional dependencies related which manipulates fuzzy functional dependencies and provide an efficient solution for the implication problem, the key findind problem, closure algorithms, redundancy removal process, etc.

**References**


