Data Dependencies in Codd's Relational Model With Similarities[*]

Radim Belohlavek[1,2] and Vilem Vychodil[2]

[1]Dept. Systems Science and Industrial Engineering, Binghamton University—SUNY
Binghamton, NY, 13902, U. S. A., rbelohla@binghamton.edu

[2]Department of Computer Science, Palacky University, Olomouc
Tomkova 40, CZ-779 00 Olomouc, Czech Republic, vilem.vychodil@upol.cz

Abstract

This chapter deals with data dependencies in Codd's relational model of data. In particular, we deal with fuzzy logic extensions of the relational model which consist in adding similarity relations to domains and consider functional dependencies in these extensions. We present a particular extension and functional dependencies in this extension which follows the principles of fuzzy logic in narrow sense. We present selected features and results regarding this extension. Then, we use this extension as a reference model and compare it to several other extensions proposed in the literature. We argue that following the principles of fuzzy logic in narrow sense the same way as following the principles of classical logic in case of ordinary Codd's relational model helps achieve transparency, versatility, conceptual clarity, and theoretical and computational tractability of the extension. We outline several topics for future research.

INTRODUCTION

Ordinary Codd's Relational Model and Data Dependencies

Codd's relational model of data is one of the most important contributions to computer science and perhaps the most important concept in data management: "A hundred years from now, I'm quite sure, database systems will still be based on Codd's relational foundation." (Date, 2000, pp. 1). Among the main virtues of the model are logical and physical data independence, access flexibility, and data integrity. They are mainly due to the reliance of the model on a simple yet powerful mathematical concept of a relation and on first-order logic: "The relational approach really is rock solid, owing (once again) to its basis in mathematics and predicate logic." (Date, 2000, pp. 138). Codd's relational model represents the theoretical foundations for relational databases.

Data dependencies represent an important tool in Codd's relational model, see, e.g. (Maier, 1983). Functional dependencies, multivalued dependencies, inclusion dependencies, and join dependencies are perhaps the most important data dependencies. They serve as constraints and are being used in the design of relational databases. Data dependencies are a classic topic in relational databases which has been thoroughly studied in the past, see (Ullman, 1988). In addition to that, data dependencies have been used for data mining purposes, see, e.g., (Manilla & Räiha, 1994).

Codd's relational model, data dependencies, and functional dependencies in particular, are the main subject of this chapter. We will now recall these notions. The central notion in Codd's model is that of a relation (table) over a relation scheme. This concept is illustrated by the table in Fig. 1. The corresponding relation scheme Y consists of attributes name, age, and education, denoted for brevity by n, a, and e. For each attribute y, the set D_y of all possible values of y is called a domain of y. For instance, D_n consists of all names (character strings). The table in Fig. 1 can be thought of as a relation $\mathcal{D}$ between domains D_n, D_a, and D_e. That is, $\mathcal{D} \subseteq D_n \times D_a \times D_e$. The tuples which belong to $\mathcal{D}$ are just the rows of the table.

[*] Supported by grant No. 1ET101370417 of GA AV ČR and by institutional support, research plan MSM 6198959214.

Figure 1: Data table over domains

name	*age*	*education*
Adams	30	Comput. Sci.
Black	30	Comput. Eng.
Chang	28	Accounting
Davis	27	Comput. Eng.
Enke	36	Electric. Eng.
Francis	39	Business

That is, ⟨Adams, 30, Comput. Sci.⟩ $\in \mathcal{D}$, ..., ⟨Francis, 39, Business⟩ $\in \mathcal{D}$, but, for instance, ⟨Adams, 30, Business⟩ $\notin \mathcal{D}$. A functional dependency (FD) over a relation scheme Y is an expression

$$A \Rightarrow B$$

where A and B are sets of attributes from Y, i.e., $A, B \subseteq Y$. A FD $A \Rightarrow B$ can be true (valid) or false (not valid) in a given table $\mathcal{D}$ over a relation scheme Y. By definition, $A \Rightarrow B$ is true in $\mathcal{D}$, denoted by $||A \Rightarrow B||_{\mathcal{D}} = 1$ iff (if and only if)

$$\begin{aligned} &\text{for any tuples } t_1, t_2 \in \mathcal{D}: \\ &\quad \text{IF } t_1, t_2 \text{ have the same values on attributes from } A \\ &\quad \text{THEN } t_1, t_2 \text{ have the same values on attributes from } B. \end{aligned} \tag{1}$$

Otherwise, i.e., if there are $t_1, t_2 \in \mathcal{D}$ such that t_1 and t_2 agree on attributes from A but disagree on some attribute from B, $A \Rightarrow B$ is not true in $\mathcal{D}$, denoted by $||A \Rightarrow B||_{\mathcal{D}} = 0$. If one thinks of $A \Rightarrow B$ as a constraint, then $||A \Rightarrow B||_{\mathcal{D}} = 1$ means that the data in $\mathcal{D}$ satisfies the constrained $A \Rightarrow B$. If we denote the fact that t_1 and t_2 have the same values on all attributes from C in table $\mathcal{D}$ by $t_1(C) =_{\mathcal{D}} t_2(C)$, we can rewrite (1) to

$$\text{for any } t_1, t_2 \in \mathcal{D}: \quad \text{IF } \ t_1(A) =_{\mathcal{D}} t_2(A) \ \text{ THEN } \ t_1(B) =_{\mathcal{D}} t_2(B). \tag{2}$$

As an example, {age} ⇒ {education} is a FD which is not true in the table in Fig. 1 because Adams and Black have the same age but they differ in their education. On the other hand, {name} ⇒ {age, education} is a FD which is true because, in our table, name uniquely determines both age and education.

It is important to note at this juncture that many of the issues related to functional dependencies can be formulated in the ordinary terms of logic. Namely, functional dependencies (and other dependencies) play the role of formulas and the data tables $\mathcal{D}$ play the role of semantic structures in which the formulas are evaluated. That is, a FD $A \Rightarrow B$ being true in $\mathcal{D}$ has the same conceptual meaning as, e.g., a first-order formula φ being true in a fist-order structure $\mathbf{M}$. Although the terminology in the theory of data dependencies differs somewhat from the standard terminology of logic, there is always a clear correspondence between the concepts in data dependencies and the ordinary concepts of logic. This certainly contributes to conceptual clarity and theoretical and computational tractability of data dependencies within the ordinary Codd's relational model.

Imprecision and Extensions of Codd's Relational Model

The foundations of relational databases have been subject to many extensions focusing on issues which, according to the authors of the extensions, are not handled appropriately by the ordinary Codd's relational model. A lot of these extensions stems from the contention, upon which there seems to be an agreement, that the original Codd's model does not provide us with adequate means for modeling of imprecision and uncertainty. As a matter of fact, management of uncertainty in data is listed among six currently most-important research directions proposed in the report from the Lowell debate by 25 senior database researchers (Abiteboul, Agrawal, Bernstein, Carey, Ceri, Croft, DeWitt, *et al.*, 2005). It was pointed out by Abiteboul, Agrawal, Bernstein, Carey, Ceri, Croft, DeWitt, *et al.* (2005) that "...current DBMS have no facilities for either approximate data or imprecise queries." Among these extensions, probabilistic extensions and fuzzy logic extensions are most common. In our chapter, we are interested in fuzzy logic extensions. For probabilistic extensions, we refer a reader to (Fuhr & Rölleke, 1997) and (Dey & Sarkar, 1996) and the references therein.

Most of the fuzzy logic extensions of Codd's relational model can be seen as attempts to develop an extension which would

$$\text{take into account } \boxed{\text{similarities}} \text{ on attribute domains.}$$

Note that the idea is quite natural and quite straightforward. Namely, it is similarity on domains which, in principle, makes it possible to consider approximate (similarity-based) queries like "select all candidates with age approximately 30". It is thus not surprising that such extensions of Codd's model appeared quite early. The first paper on this topic is (Buckles & Petry, 1982). Another stream in fuzzy extensions is motivated by the idea to allow sets or fuzzy sets of domain elements in table entries and/or queries instead of the domain elements themselves. For instance, one might want to have fuzzy sets representing terms "young", "old" etc., instead of exact values of age. This idea was for the first time studied in (Prade & Testemale, 1984). In our chapter, we are primarily interested in similarity-extensions of Codd's model and in particular, in data dependencies over these extensions. Since (Buckles & Petry, 1982), many papers on these topics have been published. We have found around 100 papers. Selected ones are listed in our references and are discussed in our paper.

Note that the ordinary Codd's model does not account for similarities because similarities are not part of Codd's model. Nevertheless, the literature contains many approaches to similarity-based querying in which an *ad hoc* similarity-module is put on top of an ordinary database. Put in this perspective, the effort represented by the existing approaches to similarity-extensions of Codd's model is to have similarities directly as a part of the model from the beginning, so that all issues including database design, relational algebra, querying, etc., be connected to similarities.

In fact, saying that similarities are not a part of the ordinary Codd's model is not accurate because Codd's model involves a very simple form of similarities, a degenerate one, so to say. Namely, Codd's model involves identities on domains. Notice that identities on domains are "behind" the exact-match-based manipulative part of relational databases, e.g., SQL commands, `SELECT`, `JOIN`, as well as "behind" the methods of reasoning about relations data, e.g., definition of validity of functional dependencies. From this standpoint, we want to replace identity relations in Codd's model by similarity relations. Intuitively, the meaning of a functional dependency $A \Rightarrow B$ being true, cf. (1), would be

$$\begin{aligned}&\text{for any tuples } t_1, t_2 \in \mathcal{D}: \\ &\quad \text{IF } t_1, t_2 \text{ have } \textit{similar values} \text{ on attributes from } A \\ &\quad \text{THEN } t_1, t_2 \text{ have } \textit{similar values} \text{ on attributes from } B.\end{aligned} \tag{3}$$

Note at this point that (3) is a conceptually new kind of dependency with a data mining appeal.

Replacing identities by similarities, is, however, not a trivial exercise. Namely,

extending Codd's model by similarities involves:

conceptual problems: Basically, a non-trivial extension of an ordinary model leads to situations where there are choices to be made on issues for which there are no choices available in the ordinary model. Needless to say, relational databases represent a complex system reaching from theoretical foundations and algorithms to implementation issues (and possibly to commercial issues). When making these choices, one needs to consider both our intentions regarding the extension, and the theoretical and computational tractability of our extension.

technical problems: The ordinary Codd's model is closely related to the concept of a relation and to ordinary first-order logic, cf. the above quotation from Date (2000, pp. 138). With similarity relations, the technical issues become more complicated. Nevertheless, one should aim at attaining the same conceptual clarity as that of the ordinary Codd's model. Since we intend to use fuzzy relations to model similarities, one should use fuzzy logic in narrow sense (i.e., a formal fuzzy logic which is an appropriate counterpart to ordinary logic) the same way ordinary logic is used in the ordinary Codd's model.

Figure 2: Ranked data table over domains with similarities

$\mathcal{D}(t)$	*name*	*age*	*education*
1.0	Adams	30	Comput. Sci.
1.0	Black	30	Comput. Eng.
0.9	Chang	28	Accounting
0.8	Davis	27	Comput. Eng.
0.4	Enke	36	Electric. Eng.
0.3	Francis	39	Business

$$n_1 \approx_n n_2 = \begin{cases} 1 & \text{if } n_1 = n_2 \\ 0 & \text{if } n_1 \neq n_2 \end{cases}$$

$$a_1 \approx_a a_2 = s_a(|a_1 - a_2|)$$
with scaling function $s_a : \mathbb{Z}^+ \to [0,1]$

$\approx_e$	A	B	CE	CS	EE
A	1	.7			
B	.7	1			
CE			1	.9	.6
CS			.9	1	.7
EE			.6	.7	1

Outline of This Chapter

The main aim of our chapter is to discuss in particular terms and using particular examples the problem of extending Codd's relational model and its data dependencies to a setting where similarities are directly a part of the model. In order to discuss in specific terms, we focus on functional dependencies within similarity extensions of Codd's model. We proceed by presenting an extension which we recently developed, see, e.g., (Belohlavek & Vychodil, 2005, 2006b, 2006c, 2006f). Basically, we will cover:

(1) **Ranked data tables over domains with similarities**. These are our counterparts to tables (relations) of the ordinary Codd's model. The concept of a ranked table over domains with similarities results from the ordinary concept of a table (relation) by adding two components:

- **Ranks**, i.e., truth degrees $\mathcal{D}(t)$ attached to the table rows (tuples) t. Doing so, our table in fact becomes a fuzzy relation between attribute domains. The basic meaning of a rank $\mathcal{D}(t)$ is: $\mathcal{D}(t)$ is a degree to which t satisfies a similarity-based constraint, such as a similarity-based query. If the table just represents stored data (i.e., prior to querying or other data manipulation), all ranks $\mathcal{D}(t)$ are equal to 1. In this case, the constraint is empty. If the table represents, e.g., a result of a query, we might have $\mathcal{D}(t) = 0.8$ or the like representing the fact that tuple t satisfies the query to degree 0.8. In this case, the constraint is represented by the query.
- **Similarities**, i.e., binary fuzzy relations $\approx_y$ on domains D_y. We require $\approx_y$ to be at least reflexive and symmetric. That is, for every $u, v \in D_y$, $u \approx_y v$ is a degree to which u is similar to v.

Fig. 2 shows a ranked table with similarities which can be thought of as an answer to a similarity-based query "select all candidates with age approximately 30". Note that $s_a : \mathbb{Z}^+ \to [0,1]$ is an appropriate scaling function assigning degrees of similarity to distances $|a_1 - a_2|$ of aga values a_1 and a_2 satisfying some natural requirements such as $s_a(0) = 1$ and $s_a(d_1) \leq s_a(d_2)$ for $d_2 \leq d_1$ and possibly other requirements.

(2) **(Fuzzy) functional dependencies**. These are our counterparts to ordinary functional dependencies. Fuzzy functional dependencies are expressions of the form

$$A \Rightarrow B$$

with both A and B being fuzzy sets of attributes, such as

$$\{{}^{1}\!/\text{make}, {}^{1}\!/\text{model}, {}^{0.7}\!/\text{year}, {}^{0.9}\!/\text{mileage}\} \Rightarrow \{{}^{0.8}\!/\text{price}\}. \tag{4}$$

If both A and B are ordinary sets, fuzzy functional dependencies become ordinary functional dependencies. If A and B are fuzzy sets, membership degrees in A and B serve as thresholds (this covers the former case as a boundary case in which the thresholds are 0 or 1 only). If tuples of our table contain information about used cars, the meaning of FD (4) would be: “for any two cars (tuples) t_1, t_2: if t_1 and t_2 are the same model of the same make, and the similarity degree of production years of t_1 and t_2 is at least 0.7, and the similarity degree of mileages of t_1 and t_2 is at least 0.9, then the price of t_1 is similar to the price of t_2 to degree at least 0.8”. Note, however, that the particular meaning of a FD $A \Rightarrow B$ depends both on the similarities on domains and on the truth functions of logical connectives, i.e., on the semantical part of our approach to FDs.

(3) **Armstrong-like axioms and completeness theorems**. We present a complete system of deduction rules, inspired by Armstrong axioms, for reasoning with FDs. We show that a complete system of deduction rules can be obtained by adding a single rule (rule of multiplication) to the ordinary Armstrong axioms where sets are replaced by fuzzy sets. We consider two types of completeness. First, the ordinary-style completeness which deals with entailment in degree 1 (full entailment) and says that $A \Rightarrow B$ semantically follows from a set T of FDs iff $A \Rightarrow B$ is provable from T. Second, Pavelka-style (graded) completeness which says that the degree to which $A \Rightarrow B$ semantically follows from T equals the degree to which $A \Rightarrow B$ is provable from T.

(4) **Further topics**. We outline: (a) a relational algebra for ranked tables over domains with similarities, (b) a link to an alternative semantics of functional dependencies which is provided by object-attribute tables with fuzzy attributes, and (c) an algorithm for computing a non-redundant basis of all FDs of a given table over domains with similarities.

The above issues are presented in Section “Ranked tables over domains with similarities and their data dependencies”. In Section “Overview of approaches to functional dependencies in Codd’s model with similarities”, we present an overview of other approaches to extensions of Codd’s model by similarities, and provide an assessment and a comparison between some of them and our approach. The following are the main points of our overview and comparison we try to emphasize.

(1) **Importance of a clear connection to fuzzy logic in narrow sense**. It has been pointed out many times, see e.g. an excellent retrospective overview by Date (2000), that a reliance of Codd’s relational model on first-order logic is among the key factors of success of Codd’s model. In case of fuzzy logic extensions of Codd’s model such as the extension by similarities, a clear connection to fuzzy logic in narrow sense is all the more important. Namely, the calculus of fuzzy logic is technically more involved than that of ordinary logic. Therefore, having a clear link of the extension to fuzzy logic in narrow sense and using it in a proper way opens a way to conceptual clarity, user friendliness of the extension, and theoretical and computational tractability of the extension. Many of the extensions of Codd’s model by similarities are, from this point of view, *ad hoc* in that a proper link to fuzzy logic in narrow sense is missing or, at least, not handled appropriately. It is our contention that this is the main reason why many of the contributions published in the literature are, by and large, “definitional” papers, i.e., papers where results demonstrating feasibility of the introduced concepts are missing. This is perhaps one of the reasons why, up to now, fuzzy logic extensions of Codd’s model did not seriously penetrate database community.

(2) **Importance of a consistent approach to extensions of Codd’s model**. Theoretical foundations of relational databases, as being represented by Codd’s model, form a complex system. All parts of Codd’s model need to be approached consistently if we want to extend Codd’s model and want to have solid foundations for relational databases with extended capabilities regarding management of uncertainty and imprecision. Here again, a clear connection to fuzzy logic in narrow sense helps manage such a consistency. It is our contention, that the state of the art in extending Codd’s model by similarities can be seen as a family of scattered approaches, often unrelated to each other, rather than a consistent collection of mutually interconnected results.

(3) **Importance of paying attention to theoretical and computational tractability**. An important feature of the ordinary Codd’s model is its theoretical and computational tractability. In many papers on extension of Codd’s model, little attention has been paid to this feature. A likely reason here, again, is a lack of clear connection to fuzzy logic in narrow sense. Note that traditional logical issues like

syntactico-semantical completeness, which is usually considered a nice theoretical property which, however, has little importance in applications, have an important practical role in Codd's model. Namely, completeness of Armstrong axioms justifies correctness of algorithms, see, e.g., (Maier, 1983). Note, however, that the area is evolving and that new publications are emerging, such as (Galindo, Urrutia, & Piattini, 2006), where computational aspects are considered.

Then, in Section "Research topics", we present several open problems in extending Codd's relational model by similarities.

PRELIMINARIES

We now recall basic notions of fuzzy logic and fuzzy set theory, for details see, e.g. (Belohlavek, 2002; Gerla, 2001, Hájek, 1998; Klir & Yuan, 1995). We pick so-called complete residuated lattices as our basic structures of truth degrees (i.e., sets of truth degrees equipped with fuzzy logic operations like implication, etc.). A complete residuated lattice is a structure $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$ where L is a set of truth degrees, $\wedge, \vee, \otimes, \rightarrow$ are operations on L, and $0, 1$ are two designated truth degrees from L. As as example, we can have $L = [0,1]$, i.e. L is a real unit interval, but in general, elements of L need not be numbers. $\wedge$ and $\vee$ are infimum and supremum on L. Note that if $L = [0,1]$, $\wedge$ and $\vee$ coincide with minimum and maximum. L equipped with $\wedge$ and $\vee$ is required to form a complete lattice. This is needed because of the semantics of the general and universal quantifiers in fuzzy logic. $\otimes$ and $\rightarrow$ are truth functions of "fuzzy conjunction" and "fuzzy implication". Although we have many choices of $\otimes$ and $\rightarrow$ (see below), the choice of $\otimes$ and $\rightarrow$ cannot be arbitrary. $\otimes$ and $\rightarrow$ need to satisfy certain properties and certain relationships, such as the adjointness property (see below), need to be satisfied between $\otimes$ and $\rightarrow$. These properties enable us to properly extend to a fuzzy setting various results from a crisp setting. Note also that the properties and relationships imposed by the concept of a complete residuated lattice are quite natural and not restrictive.

Formally, a complete residuated lattice is an algebra $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$ such that

(i) $\langle L, \wedge, \vee, 0, 1\rangle$ is a complete lattice with 0 and 1 being the least and greatest element of L, respectively;

(ii) $\langle L, \otimes, 1\rangle$ is a commutative monoid (i.e., $\otimes$ is commutative, associative, and for each $a \in L$ we have $a \otimes 1 = 1 \otimes a = a$);

(iii) $\otimes$ and $\rightarrow$ satisfy so-called adjointness property:

$$a \otimes b \leq c \quad \text{iff} \quad a \leq b \rightarrow c \tag{5}$$

for each $a, b, c \in L$.

Throughout this paper, $\mathbf{L}$ denotes an arbitrary complete residuated lattice. In addition to that, we consider so-called (truth-stressing) hedges, i.e. unary operations on L satisfying

$$1^* = 1, \tag{6}$$

$$a^* \leq a, \tag{7}$$

$$(a \rightarrow b)^* \leq a^* \rightarrow b^*, \tag{8}$$

$$a^{**} = a^*, \tag{9}$$

for each $a, b \in L$. Elements $a \in L$ are called a truth degrees; $\otimes$ and $\rightarrow$ are (truth functions of) "fuzzy conjunction" and "fuzzy implication"; * is a (truth function of) logical connective "very true" and properties of hedges have natural interpretations, see (Hájek, 1998; Hájek, 2001).

A common choice of $\mathbf{L}$ is a structure with $L = [0,1]$ (unit interval), $\wedge$ and $\vee$ being minimum and maximum, $\otimes$ being a left-continuous t-norm with the corresponding $\rightarrow$. Three most important pairs of adjoint operations on the unit interval are:

(i) Łukasiewicz: $a \otimes b = \max(a+b-1, 0)$; $a \rightarrow b = \min(1-a+b, 1)$;

(ii) Gödel: $a \otimes b = \min(a,b)$; $a \rightarrow b = 1$ if $a \leq b$, $a \rightarrow b = b$ else;

(iii) Goguen (product): $a \otimes b = a \cdot b$; $a \rightarrow b = 1$ if $a \leq b$, $a \rightarrow b = \frac{b}{a}$ else.

Complete residuated lattices include also finite structures of truth degrees (e.g., finite Łukasiewicz and Gödel chains). Two boundary cases of hedges are

(i) identity, i.e. $a^* = a$ $(a \in L)$;

(ii) so-called globalization (Takeuti & Titani, 1987):

$$a^* = \begin{cases} 1 & \text{if } a = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

A special case of a complete residuated lattice with hedge is the two-element Boolean algebra $\mathbf{2} = \langle \{0,1\}, \wedge, \vee, \otimes, \rightarrow, {}^*, 0, 1 \rangle$, i.e. the structure of truth degrees of classical logic.

An $\mathbf{L}$-set (a fuzzy set) A in universe U is a mapping $A\colon U \rightarrow L$, $A(u)$ being interpreted as "the degree to which u belongs to A". $\mathbf{L}^U$ denotes the collection of all $\mathbf{L}$-sets in U. The operations with $\mathbf{L}$-sets are defined componentwise. For instance, union of $\mathbf{L}$-sets $A, B \in \mathbf{L}^U$ is an $\mathbf{L}$-set $A \cup B$ in U such that $(A \cup B)(u) = A(u) \vee B(u)$ $(u \in U)$. Binary $\mathbf{L}$-relations (binary fuzzy relations) in U are just fuzzy sets in $U \times U$. An $\mathbf{L}$-set A in U is called crisp if $A(u) = 0$ or $A(u) = 1$ for each $u \in U$. Obviously, crisp $\mathbf{L}$-sets in U are precisely the characteristic functions of ordinary subsets of U. Therefore, we will identify crisp $\mathbf{L}$-sets in U with ordinary subsets of U.

RANKED TABLES OVER DOMAINS WITH SIMILARITIES AND THEIR DATA DEPENDENCIES

Ranked tables over domains with similarities

The main motivation for ranked tables over domains with similarities is the fact that for many domains, it is desirable to consider degrees of similarity of their elements rather than only "equal" and "not equal". We introduced ranked tables over domains with similarities in (Belohlavek & Vychodil, 2005, 2006b, 2006c). However, the idea of equipping domains with similarity relations goes back to the early approaches to Codd's model from the point of view of fuzzy logic, particularly to (Buckles & Petry, 1982). The first paper which considers both similarities and ranks is probably Raju and Majumdar's (1988) which is one of the most advanced and influential papers on fuzzy extensions of Codd's model. However, the meaning of ranks is intuitively not quite clear in (Raju & Majumdar, 1988). While we interpret a rank assigned to a tuple as a degree to which the tuple matches a similarity-based constraint such as a similarity-based query, see Section "Outline of This Chapter", Raju and Majumdar describe a rank as a degree to which a tuple belongs to a table. Later on, in Example 3.1, they say that a rank can be interpreted as a possibility measure or a measure of association of the items of a tuple. As we will see later, Raju and Majumdar's extension of Codd's model including their concept of a functional dependency is a particular case of our extension. Another paper with similar ideas is (Medina, Pons, & Villa, 1994). Note, however, that various further ways of using degrees instead of just 0 and 1 have been studied in the literature, see e.g. (Galindo, Urrutia, & Piattini, 2006).

The concept of a ranked table over domains with similarities is depicted in Fig. 2. It consists of three parts: data table (relation), domain similarities, and ranking. The data table (right table in the upper part of Fig. 2) coincides with a data table of the ordinary relational model. Domain similarities and ranking are what makes our model an extension of the ordinary model. The domain similarities (bottom part of Fig. 2) assign degrees of similarity to pairs of values of the respective domain. For instance, a degree of similarity of "Computer Science" and "Computer Engineering" is 0.9 while a degree of similarity of "Computer Science" and "Electrical Engineering" is 0.6. The ranking assigns to each row (tuple) of the data table a degree of a scale bounded by 0 and 1 (left top table in Fig. 2), e.g. 0.9 assigned to the tuple $\langle \text{Chang}, 28, \text{Accounting} \rangle$. In general, a rank represents a degree to which the corresponding tuple satisfies a given similarity-based query (rank = degree to which a tuple matches a query). For instance, the ranked table of Fig. 2 can result as an answer to query "select all candidates with age approximately 30". In a data table representing stored data (i.e. prior to any querying), ranks of all tuples of the table are equal to 1. This corresponds to a situation where the similarity-query does not impose any constraint, i.e., the query reads “select all tuples”. Therefore, the same way as tables in the classical relational model, ranked tables represent both stored data and outputs to queries. This is an important feature of our model.

A formal definition follows (Y denotes a set of attributes, attributes are denoted by $y, y_1, \ldots$; $\mathbf{L}$ denotes a fixed structure of truth degrees).

Definition 1. A *ranked* $\boxed{\textit{data table}}$ *$\mathcal{D}$ over domains with* $\boxed{\textit{similarity}}$ *relations* (with Y and $\mathbf{L}$) is given by

- *domains*: for each $y \in Y$, D_y is a non-empty set (domain of y, set of values of y);
- *similarities*: for each $y \in Y$, $\approx_y$ is a binary fuzzy relation (called similarity) in D_y (i.e., a mapping $\approx_y$: $D_y \times D_y \to L$) which is reflexive ($u \approx_y u = 1$ for each $u \in D_y$) and symmetric ($u \approx_y v = v \approx_y u$ for all $u, v \in D_y$);
- *ranking*: for each tuple $t \in \times_{y \in Y} D_y$, there is a degree $\mathcal{D}(t) \in L$ (called rank of t in $\mathcal{D}$) assigned to t.

Remark 1. (1) $\mathcal{D}$ can be seen as a table with rows and columns corresponding to tuples and attributes, like in Fig. 2. By $t[y]$ we denote a value from D_y of tuple t on attribute y. We require that there is only a finite number of tuples which get assigned a non-zero degree (i.e., the corresponding table is finite). Clearly, if $L = \{0, 1\}$ and if each $\approx_y$ is equality, the concept of a ranked data table with similarities coincides with that of a data table (relation) of a classical Codd’s model.

(2) Formally, $\mathcal{D}$ is a $\boxed{\text{fuzzy relation}}$ between domains D_y ($y \in Y$). As mentioned above, $\mathcal{D}(t)$ is interpreted as a degree to which the tuple t satisfies constraints posed by a similarity-based query. We say “non-ranked table” if for each tuple t, $\mathcal{D}(t) = 0$ or $\mathcal{D}(t) = 1$. This accounts for tables representing stored data (prior to querying).

(3) One can require additional properties of $\approx_y$, such as transitivity w.r.t. (with respect to) a binary operation $\odot$ on L, i.e., $(u \approx_y v) \odot (v \approx_y w) \leq (u \approx_y w)$, or separability, i.e., $u \approx_y v = 1$ iff $u = v$, which is sometimes required in the literature. We are not concerned here with how the similarities are represented (we assume they can either be computed or, if D_y is small, are stored).

(4) It is interesting to note that ranked tables over domains with similarities are implicitly used in recent papers in database community, e.g., in (Illyas, Aref, & Elmagarmid, 2004) and (Li, Chang, Ilyas, & Song, 2005).

Functional dependencies in ranked tables over domains with similarities

Functional dependencies (FDs) are the most studied data dependencies within fuzzy logic extensions of Codd’s model. To keep our paper focused, we deal only with functional dependencies. Note, however, that other types of dependencies have been investigated as well, see, e.g., (Liu, 1997; Sharma, Goswami, & Gupta, 2004). In the context of ranked tables over domains with similarities, we introduced FDs as follows (Belohlavek & Vychodil, 2005, 2006b):

Definition 2. A (*fuzzy*) $\boxed{\textit{functional dependency}}$ is a formula $A \Rightarrow B$ where A and B are fuzzy sets of attributes ($A, B \in \mathbf{L}^Y$).

We first present a definition of validity of $A \Rightarrow B$ in a ranked data table $\mathcal{D}$ and then add comments.

Definition 3. For a ranked data table $\mathcal{D}$, tuples t_1, t_2 and a fuzzy set $C \in \mathbf{L}^Y$ of attributes, we introduce a *degree $t_1(C) \approx_{\mathcal{D}} t_2(C)$ to which t_1 and t_2 have similar values on attributes from C* by

$$\begin{aligned} &t_1(C) \approx_{\mathcal{D}} t_2(C) = \\ &= (\mathcal{D}(t_1) \otimes \mathcal{D}(t_2)) \to \textstyle\bigwedge_{y \in Y}(C(y) \to (t_1[y] \approx_y t_2[y])). \end{aligned} \tag{11}$$

A *degree $||A \Rightarrow B||_{\mathcal{D}}$ to which a FD $A \Rightarrow B$ is true in $\mathcal{D}$* is defined by

$$\begin{aligned} &||A \Rightarrow B||_{\mathcal{D}} = \\ &= \textstyle\bigwedge_{t_1, t_2} \big((t_1(A) \approx_{\mathcal{D}} t_2(A))^* \to (t_1(B) \approx_{\mathcal{D}} t_2(B))\big). \end{aligned} \tag{12}$$

Remark 2. (1) One can easily see the following: if both A and B are crisp, $A \Rightarrow B$ is just an ordinary FD; if, moreover, each similarity $\approx_y$ is an identity, then $||A \Rightarrow B||_{\mathcal{D}} = 1$ iff $A \Rightarrow B$ is true in the ordinary data table corresponding to $\mathcal{D}$ in the usual sense. Note that $\bigwedge$ denotes infimum, see Section "Preliminaries".

(2) By basic rules of semantics of fuzzy logic in narrow sense (Hájek, 1998), $t_1(C) \approx_{\mathcal{D}} t_2(C)$ is just the truth degree of formula "*if t_1, t_2 are from $\mathcal{D}$ then for each attribute y from C, t_1 and t_2 have similar values on y*". Moreover, $||A \Rightarrow B||_{\mathcal{D}}$ is a truth degree of a formula "*for any tuples t_1, t_2: if t_1 and t_2 have similar values on attributes from A then t_1 and t_2 have similar values on attributes from B*". Note that due to our adherence to predicate fuzzy logic, the meaning of $A \Rightarrow B$ is given by a simple formula which we just described in natural language. Note that, in fact, the antecedent in formula (12) is modified by a hedge *. This has technical reasons not discussed in detail here. Note only that setting * to globalization or identity enables us to have some of previous approaches which can be found in the literature as particular cases of our approach, see Section "Overview of approaches to functional dependencies in Codd's model with similarities".

(3) Note also that $||A \Rightarrow B||_{\mathcal{D}}$ is a truth degree from our scale L, not necessarily being 0 or 1, and that this comes up naturally in the context of fuzzy logic in narrow sense. That is, our FDs may be true to a degree, e.g., 0.9 (approximately true) which is natural when considering approximate concepts like similarity. The particular value and the meaning of $||A \Rightarrow B||_{\mathcal{D}}$ depends on our choice of the scale of truth degrees and the connectives. For illustration, suppose that the ranks in $\mathcal{D}$ are all 0 or 1, i.e., $\mathcal{D}$ represents stored data. If * is globalization, then for any choice of the scale L and connectives $\otimes, \rightarrow$ we have that $||A \Rightarrow B||_{\mathcal{D}} = 1$ ($A \Rightarrow B$ is fully true in $\mathcal{D}$) iff for every tuples t_1, t_2 from $\mathcal{D}$: if $A(y) \leq (t_1[y] \approx_y t_2[y])$ for any attribute $y \in Y$ then $B(y) \leq (t_1[y] \approx_y t_2[y])$ for any attribute $y \in Y$. This also shows that degrees $A(y)$ and $B(y)$ serve basically as similarity thresholds. If, on the other hand, * is the identity on L and both A and B are ordinary sets, then for any choice of the scale L and connectives $\otimes, \rightarrow$ we have that $||A \Rightarrow B||_{\mathcal{D}} = 1$ iff $\bigwedge_{y \in A}(t_1[y] \approx_y t_2[y]) \leq \bigwedge_{y \in B}(t_1[y] \approx_y t_2[y])$. Therefore, different choices of truth functions of logical connectives lead to different "numerical interpretations" of $A \Rightarrow B$ being fully true in $\mathcal{D}$. However, the essential meaning of $A \Rightarrow B$ being fully true in $\mathcal{D}$ remains the same. Namely, the essential meaning is given by the formula "*for any tuples t_1, t_2: if t_1 and t_2 have similar values on attributes from A then t_1 and t_2 have similar values on attributes from B*" and this formula remains the same. Note that this is a characteristic feature of fuzzy logic: a single formula (representing syntactically a meaning of a proposition) can lead to different truth values and thus different conditions under which this formula is true.

Armstrong-like axioms and completeness

We now turn our attention to a classic problem in FDs, namely, axiomatizability by Armstrong axioms. First, we introduce the necessary semantical concepts related to fuzzy FDs. For a set T of fuzzy FDs, let $\text{Mod}(T)$ be a set of all ranked data tables with similarities in which each FD from T is true in degree 1, i.e.

$$\text{Mod}(T) = \{\mathcal{D} \mid \text{ for each } A \Rightarrow B \in T : ||A \Rightarrow B||_{\mathcal{D}} = 1\}.$$

$\mathcal{D} \in \text{Mod}(T)$ are called models of T. A *degree $||A \Rightarrow B||_T$ to which $A \Rightarrow B$ semantically follows from T* is defined by

$$||A \Rightarrow B||_T = \textstyle\bigwedge_{\mathcal{D} \in \text{Mod}(T)} ||A \Rightarrow B||_{\mathcal{D}}$$

where the infimum ranges over all models of T. Note that $||A \Rightarrow B||_T$ can be interpreted as a truth degree of "for each model $\mathcal{D}$ of T: $A \Rightarrow B$ is true in $\mathcal{D}$". Note again that the previous concepts are based on the principles of fuzzy logic in narrow sense, see (Hájek, 1998). As a result, the concepts have a clear meaning and obey general rules of fuzzy logic in narrow sense.

Consider now an axiomatic system consisting of the following deduction rules:

(Ax) infer $A \cup B \Rightarrow A$,

(Cut) from $A \Rightarrow B$ and $B \cup C \Rightarrow D$ infer $A \cup C \Rightarrow D$,

(Mul) from $A \Rightarrow B$ infer $c^* \otimes A \Rightarrow c^* \otimes B$

for each $A, B, C, D \in \mathbf{L}^Y$, and $c \in L$.

Remark 3. (1) Rules (Ax), (Cut), and (Mul) are called the rules of axiom, cut, and multiplication. Note that (Ax) and (Cut) are well-known ordinary rules. Namely, with A, B, C, D being ordinary sets, (Ax) and (Cut) form a syntactico-semantically complete system for reasoning with ordinary FDs. (Mul) is a new rule. Note that $c^* \otimes A \in \mathbf{L}^Y$ is defined by $(c^* \otimes A)(y) = c^* \otimes A(y)$.

As usual, $A \Rightarrow B$ is called *provable* from a set T of FDs, written $T \vdash A \Rightarrow B$, if there is a sequence $\varphi_1, \ldots, \varphi_n$ of FDs such that φ_n is $A \Rightarrow B$ and for each φ_i we either have $\varphi_i \in T$ or φ_i is inferred (in one step) from some of the preceding FDs (i.e., $\varphi_1, \ldots, \varphi_{i-1}$) using some deduction rule (Ax)–(Mul).

The following theorem shows that the concept of provability of FDs coincides with the concept of semantic entailment of fuzzy FDs in degree 1, see (Belohlavek & Vychodil 2006b, 2006c):

Theorem 1 (completeness)**.** *Let T be a set of FDs, $\mathbf{L}$ and Y be finite. For each $A \Rightarrow B$ we have*

$$T \vdash A \Rightarrow B \quad \textit{iff} \quad ||A \Rightarrow B||_T = 1.$$

Remark 4. (1) Theorem 1 generalizes the ordinary completeness of Armstrong axioms (Armstrong, 1974; Maier, 1983). Our results "become" the ordinary ones if we take the two-element Boolean algebra for our structure of truth degrees. Namely, then, (Ax) and (Cut) become ordinary deduction rules which are known to be complete w.r.t. the semantics given by the ordinary Codd's model. (Mul) either becomes a trivial rule "from $A \Rightarrow B$ infer $A \Rightarrow B$" if $c = 1$ or a superfluous rule "from $A \Rightarrow B$ infer $\emptyset \Rightarrow \emptyset$".

(2) Note also that the completeness theorem holds true for an arbitrary structure $\mathbf{L}$ of truth degrees; in fact, a finite one, but infinite can be handled too, see (Belohlavek & Vychodil, 2006d).

Notice that semantic entailment is not captured entirely in Theorem 1. Namely, Theorem 1 provides a syntactic description of entailment in degree 1 only, i.e., of full entailment of $A \Rightarrow B$ from T. Other degrees of entailment, i.e., cases $||A \Rightarrow B||_{\mathcal{D}} \neq 1$, elude Theorem 1. From the point of view of fuzzy logic, this is unfortunate because Theorem 1 makes a sharp distinction: $||A \Rightarrow B||_{\mathcal{D}} = 1$ is captured but $||A \Rightarrow B||_{\mathcal{D}} = 0.9$ is not by Theorem 1. Extending the ordinary notion of provability to that of a degree of provability, i.e., from a bivalent one to a graded one, goes back to seminal work of Pavelka (1979), see also (Gerla, 2001), (Hájek, 1998), and (Novák, Perfilieva, & Močkoř, 1999). Without going to details we now show how the concept of degree of provability can be used in our setting of fuzzy FDs. For details, we refer to (Belohlavek & Vychodil, 2005, 2006b).

For a set T of fuzzy FDs and a FD $A \Rightarrow B$, let us introduce a *degree $|A \Rightarrow B|_T \in L$ to which $A \Rightarrow B$ is provable from T* by

$$|A \Rightarrow B|_T = \bigvee \{c \in L \,|\, T \vdash A \Rightarrow c \otimes B\}.$$

That is, the concept of a degree of provability is defined via the concept of an ordinary provability. The following theorem shows that the above-introduced concept of a degree of provability coincides with that of a degree of entailment, see (Belohlavek & Vychodil, 2006b, 2006c):

Theorem 2 (graded completeness)**.** *Let T be a set of FDs, $\mathbf{L}$ and Y be finite. For each $A \Rightarrow B$ we have*

$$|A \Rightarrow B|_T = ||A \Rightarrow B||_T.$$

Note that Theorem 2 can be generalized for fuzzy sets T of formulas (i.e., for reasoning from partially true premisses) and that we can also have a so-called Pavelka-style logic for sound and complete reasoning with fuzzy FDs, see (Belohlavek & Vychodil, 2006e).

Further topics in ranked tables over domains with similarities

Relational algebra and calculus

Relational algebra $\boxed{\text{Relational algebra}}$ of the ordinary model is based on the calculus of ordinary relations. In the same spirit, since ranked tables are in fact fuzzy relations, relational algebra of ranked tables is based on the calculus of fuzzy relations. Due to the limited scope, we present just an outline of the algebra for ranked tables. The algebra is relative to $\mathbf{L}$ (i.e., $\mathbf{L}$ is a parameter). Operations of the algebra can be classified as follows.

Figure 3: Illustration of similarity-based join

$\mathcal{D}(t)$	*position*	*education*
1.0	programmer	Comput. Sci.
1.0	syst. technician	Comput. Eng.

$\mathcal{D}(t)$	*name*	*position*
1.0	Adams	programmer
1.0	Black	syst. technician
0.9	Adams	syst. technician
0.9	Black	programmer

Counterparts to Boolean operations of classical model Here, for any binary (and similar for other arities) operation $\odot$ with fuzzy relations, we define a corresponding operation (denoted again) $\odot$ which yields for any two ranked tables $\mathcal{D}_1$ and $\mathcal{D}_2$ (with common Y, domains, and similarities) a ranked table $\mathcal{D}$ assigning to any tuple t a rank $\mathcal{D}(t)$ defined componentwise by

$$\mathcal{D}(t) = \mathcal{D}_1(t) \odot \mathcal{D}_2(t).$$

This accounts for operations based on $\wedge$, $\vee$, $\otimes$, $\rightarrow$. For example, with $\wedge$ and $\vee$, we obtain the counterpart of intersection and union. Note that, one has to be careful when reducing operations to other operations. For instance, unlike classical case, De Morgan law is not available in fuzzy logic in general (availability of De Morgan law depends on the negation connective we use). As a consequence, union cannot generally be expressed by intersection and complement.

New operations based on calculus of fuzzy relations The calculus of fuzzy relations contains operations which either have no counterparts with classical relations or the counterparts are trivial. An interesting example is a so-called a-cut of a fuzzy relation. For a ranked table $\mathcal{D}$ and a rank $a \in L$, an a-cut of $\mathcal{D}$ is a ranked table ${}^{a}\mathcal{D}$ defined by

$$[{}^{a}\mathcal{D}](t) = \begin{cases} 1 & \text{if } \mathcal{D}(t) \geq a, \\ 0 & \text{otherwise.} \end{cases}$$

That is, ${}^{a}\mathcal{D}$ is a non-ranked table which contains those tuples of $\mathcal{D}$ with ranks greater or equal to a. This is quite a natural operation for manipulation of ranked tables which allows the user to select only a part of a query result given by threshold a. Note that in combination with intersection, a-cut is able to keep the original ranks. Namely, we have $[\mathcal{D} \wedge {}^{a}\mathcal{D}](t) = \mathcal{D}(t)$ if $\mathcal{D}(t) \geq a$ and $= 0$ otherwise.

*Counterparts to selection, join, projection, division*These operation stem basically from the ordinary ones by taking into account similarity relations (or, in general fuzzy relations θ in place of ordinary comparators). For illustration, we consider a similarity-based join. For simplicity, consider a ranked table $\mathcal{D}_1$ from Fig. 2 (result to a query "…candidates with age approximately 30") and a ranked table $\mathcal{D}_2$ from Fig. 3 (top) describing open positions with required education. A similarity-based join $\mathcal{D}_1 \bowtie \mathcal{D}_2$ then describes possible job assignments. A rank $[\mathcal{D}_1 \bowtie \mathcal{D}_2](n,a,e,p)$ of tuple $\langle n,a,e,p \rangle$ in $\mathcal{D}_1 \bowtie \mathcal{D}_2$ is given by

$$\textstyle\bigvee_{e_1,e_2}(\mathcal{D}_1(n,a,e_1) \otimes (e_1 \approx_e e) \otimes (e \approx_e e_2) \otimes \mathcal{D}_2(p,e_2))$$

where e_1, e_2 range over the domain corresponding to *education*. That is, the join runs not only over equal values but also over similar values at the cost of decreasing the value of the resulting tuples by degrees of similarity. The bottom table of Fig. 3 shows a result of an intersection of 0.9-cut of $\mathcal{D}_1 \bowtie \mathcal{D}_2$ with $\mathcal{D}_1 \bowtie \mathcal{D}_2$, projected to *name* and *position*.

Further non-classical operations Among these operations are operations interesting from the point of information retrieval which cannot be accounted for in the ordinary model. As an example, consider top_k which gained a considerable interest recently, see (Fagin, 1999, 2002) and also (Illyas, Aref, & Elmagarmid, 2004). In the chapter by Mouaddib, Raschia, Ughetto and Voglozin, the reader can find a discussion about the computation of top_k. We define $\text{top}_k(\mathcal{D})$ to contain the first k tuples (according to rank ordering) of

$\mathcal{D}$ with their ranks (if there are less than k ranks in $\mathcal{D}$ then $\mathrm{top}_k(\mathcal{D}) = \mathcal{D}$; and $\mathrm{top}_k(\mathcal{D}))$ includes also the tuples with rank equal to the rank of the k-th tuple). Note that top_k is a part of a query language described in Penzo (2005).

Tuple and domain relational calculi The tuple calculus of the ordinary model is based on the ordinary predicate logic. In the same spirit (here again, as with relational algebra), the tuple calculus for ranked tables over domains with similarities is based on fuzzy predicate logic. It is important for our purpose that predicate fuzzy logic(s) are developed nowadays and that they are in a relationship to the calculus of fuzzy relations which is analogous to the relationship of the ordinary predicate logic to the calculus of classical relations. Expressions of our tuple calculus are of the form

$$\{x(R) \mid f(x)\}$$

with the usual meaning of the components (x the only free variable in a legal formula f, R a set of attributes). Formulas $f(x)$ are built from atoms using symbols of connectives of fuzzy logic in the usual way. In addition to this, atoms include truth constants $a \in L$, and we need a unary connective Δ (Baaz's delta). The truth function of Δ on $L = [0,1]$ assigns 1 to 1 and assigns 0 to any degree different from 1. See (Hájek, 1998) for. We have also non-standard quantifiers (Hájek, 1998) in our language like $Q_{<k}$ ("less than k") with $(Q_{<k}x)f(x)$ having truth degree 1 if the number of tuples for which $f(x)$ evaluates to a non-zero degree is less than k and having truth degree 0 otherwise. Due to inclusion of $Q_{<k}$, tuple calculus has expressions equivalent to top_k, one of them being a formula

$$\mathcal{D}(x) \wedge (Q_{<k}y)(\neg\Delta(\mathcal{D}(y) \to \mathcal{D}(x)) \wedge \Delta(\mathcal{D}(x) \to \mathcal{D}(y))).$$

The situation is similar for a domain relational calculus. Taking appropriate care of the details, one can obtain the following theorem.

Theorem 3 (equivalence theorem)**.** Our $\boxed{\text{relational algebra}}$, domain calculus, and tuple calculus are mutually equivalent.

That is, for any expression E_A of our relational algebra there is an expression E_D of our domain calculus such that for any state of any database d, the ranked tables $E_A(d)$ and $E_D(d)$, to which E_A and E_D evaluate, coincide and *vice versa* (and the same for the other cases).

Remark 5. Previous approaches either consider only similarities (Buckles, Petry & Sachar, 1989) or only ranks (Takahashi, 1993) but not both. Most importantly, our approach provides more expressive power (including, e.g., top_k) and a firm connection to predicate fuzzy logic due to which both the relational algebra and calculi are open for further extensions (e.g., by other non-standard quantifiers, aggregation operators, etc.). Li, Chang, Ilyas, & Song (2005) present an interesting framework different from ours but with similar aims.

Alternative semantics: a link to attribute implications over fuzzy attributes

Fuzzy FDs have an alternative semantics. Namely, they can be interpreted, i.e., evaluated, in data tables with fuzzy attributes. Tables with fuzzy attributes represent the input data in formal concept analysis, see (Ganter & Wille, 1999). A table with fuzzy attributes is given by a triplet $\langle X,Y,I\rangle$ consisting of a set X (so-called objects, labels of table rows), a set Y (attributes, labels of table columns), and a fuzzy relation I between X and Y with $I(x,y) \in L$ being interpreted as a degree to which object x has (fuzzy) attribute y. Given $\langle X,Y,I\rangle$, one can define a degree $||A \Rightarrow B||_{\langle X,Y,I\rangle}$ to which a fuzzy FD $A \Rightarrow B$ (called a fuzzy attribute implication in the setting of formal concept analysis) is true (valid) in $\langle X,Y,I\rangle$. Without going into details, $||A \Rightarrow B||_{\langle X,Y,I\rangle}$ is a truth degree of "for each object (row) $x \in X$: if x has all attributes from A then x has all attributes from B". A crucial fact, which provides a useful connection between these two kinds of semantics for fuzzy FDs, is that that the semantic entailment w.r.t. the semantics given by ranked tables over domains with similarities coincides with semantic entailment w.r.t. the semantics given by tables with fuzzy attributes, see Belohlavek & Vychodil (2005). This relationship has important consequences. For instance, any axiomatic system which is complete w.r.t. one semantics is also complete w.r.t. the other semantics. For more details we refer to an overview paper (Belohlavek & Vychodil, 2006a).

Non-redundant bases

In this section, we focus on non-redundant bases of FDs of ranked data tables, i.e., minimal sets T of FDs which are fully true in a given ranked table $\mathcal{D}$ such that any other FD true in $\mathcal{D}$ follows semantically from T. Non-redundant bases are therefore minimal sets of FDs which convey information about all FDs which are fully true in a given ranked table. The interest in obtaining non-redundant bases is basically twofold. First, from the point of view of knowledge extraction, a ranked data table $\mathcal{D}$ represents an answer to a similarity-based query. A non-redundant basis of $\mathcal{D}$ thus represents an additional information to the query which describes all dependencies satisfied by the result to the query. Second, as in the ordinary case, non-redundant sets of FDs are important in considerations concerning data redundancy and normalization (this applies particularly to non-ranked tables).

Computational aspects of fuzzy approaches to FDs are scarcely discussed in the literature and (Wang, Tsai & Hong, 2000) seems to be an exception. However, since the aim of Wang, Tsai & Hong (2000) is different from computing non-redundant bases, we do not discuss it here (in (Wang, Tsai & Hong, 2000), the authors compute *all* FDs satisfying some non-triviality conditions). We now present a couple of results related to the problem computing non-redundant bases of ranked data tables. For details, we refer to (Belohlavek & Vychodil, 2006b, 2006c). Let $\mathcal{D}$ be a ranked data table with similarities.

Definition 4. A set T of FDs is *complete in $\mathcal{D}$* if, for each $A \Rightarrow B$, $||A \Rightarrow B||_T = ||A \Rightarrow B||_{\mathcal{D}}$. Moreover, if T is complete in $\mathcal{D}$ and no proper subset of T is complete in $\mathcal{D}$, we call T a *non-redundant basis of $\mathcal{D}$.* T is called a *minimal basis of $\mathcal{D}$* if T is complete in $\mathcal{D}$ and for each T' which is complete in $\mathcal{D}$ we have $|T| \leq |T'|$.

We now proceed in two steps: First, we define a special closure operator $\mathrm{C}_{\mathcal{D}}$ which assigns to any fuzzy set A of attributes its closure $\mathrm{C}_{\mathcal{D}}(A)$, which is again a fuzzy set of attributes, so that $T = \{A \Rightarrow \mathrm{C}_{\mathcal{D}}(A) \,|\, A \in \mathbf{L}^Y\}$ is complete in $\mathcal{D}$. Second, we describe a "small subset" of T which is non-redundant (and minimal in size in some important cases) and computationally tractable.

Definition 5. For a ranked data table $\mathcal{D}$ over attributes Y define an operator $\mathrm{C}_{\mathcal{D}}\colon \mathbf{L}^Y \to \mathbf{L}^Y$ by

$$\begin{aligned}(\mathrm{C}_{\mathcal{D}}(A))(y) &= \\ &= \textstyle\bigwedge_{t,t'}((\mathcal{D}(t) \otimes \mathcal{D}(t') \otimes (t(A) \approx t'(A))^*) \to (t[y] \approx_y t'[y])).\end{aligned}$$

Observe that the tuples t for which $\mathcal{D}(t) = 0$ can be disregarded in the formula for $\mathrm{C}_{\mathcal{D}}$. In words, $(\mathrm{C}_{\mathcal{D}}(A))(y)$ is the degree to which "for every tuples t,t' from $\mathcal{D}$, if t and t' agree on A then they agree on y".

Theorem 4. *For each $\mathcal{D}$, $\mathrm{C}_{\mathcal{D}}$ is a closure operator, and $T = \{A \Rightarrow \mathrm{C}_{\mathcal{D}}(A) \,|\, A \in \mathbf{L}^Y\}$ is complete in $\mathcal{D}$.*

We now focus on finding a non-redundant basis of $\mathcal{D}$ which is a subset of the set T described in Theorem 4. We take advantage of the technical concept of a system of pseudo-closed fuzzy sets of attributes. Given $\mathcal{D}$, a collection $\mathcal{P} \subseteq \mathbf{L}^Y$ of fuzzy sets of attributes is called a *system of pseudo-closed fuzzy sets w.r.t. $\mathcal{D}$* if, for each $P \in \mathbf{L}^Y$, we have:

$$\begin{aligned}P \in \mathcal{P} \;\text{ iff }\; & P \neq \mathrm{C}_{\mathcal{D}}(P) \text{ and for each } Q \in \mathcal{P} \\ & \text{such that } Q \neq P\colon\; S(Q,P)^* \leq S(\mathrm{C}_{\mathcal{D}}(Q),P),\end{aligned}$$

where "$S(\cdot,\cdot)$" denote degrees of subsethood defined by

$$S(Q,P) \;=\; \textstyle\bigwedge_{y \in Y}(Q(y) \to P(y)).$$

Each $P \in \mathcal{P}$ is then called a *pseudo-closed fuzzy set of attributes.* Then, one can prove

Theorem 5. *If $\mathcal{P}$ is a system of pseudo-closed fuzzy sets w.r.t. $\mathcal{D}$, then $T = \{P \Rightarrow \mathrm{C}_{\mathcal{D}}(P) \,|\, P \in \mathcal{P}\}$ is a non-redundant basis of $\mathcal{D}$. If * is globalization, then T is a minimal basis of $\mathcal{D}$.*

Figure 4: Illustrative data table: power consumption of countries with very large populations

$\mathcal{D}(t)$
1.0
1.0
0.6
0.3
0.3
0.2
0.2
0.2
0.1

country	*coal*	*air*	*water*	*nuclear*
China	498.0	246	196	34.6
India	154.3	1032	75	26.8
USA	570.7	2533	330	753.9
Russia	115.8	54	157	122.5
Japan	0.0	120	90	293.8
Germany	56.4	3817	50	161.2
UK	19.5	350	8	81.7
France	0.0	63	62	394.4
Spain	10.9	1180	11	58.9

Figure 5: Illustrative data table: particular similarity relations on domains

$\approx_c$	Cn	In	US	Ru	Jp	Ge	Fr	UK	Sp
Cn	1		.3						
In		1		.6					
US	.3		1						
Ru		.6		1		.4			
Jp					1	.4	1	.8	.9
Ge				.4	.4	1	.4	.7	.6
Fr					1	.4	1	.8	.9
UK					.8	.7	.8	1	1
Sp					.9	.6	.9	1	1

$\approx_a$	Cn	In	US	Ru	Jp	Ge	Fr	UK	Sp
Cn	1	.5		.9	.9		.9	.9	.4
In	.5	1		.3	.4		.3	.5	.9
US			1			.1			.1
Ru	.9	.3		1	1		1	.8	.2
Jp	.9	.4		1	1		1	.9	.3
Ge			.1			1			
Fr	.9	.3		1	1		1	.8	.2
UK	.9	.5		.8	.9		.8	1	.4
Sp	.4	.9	.1	.2	.3		.2	.4	1

$\approx_w$	Cn	In	US	Ru	Jp	Ge	Fr	UK	Sp
Cn	1			.6					
In		1			1	.9	1	.2	.2
US			1						
Ru	.6			1	.2				
Jp		1		.2	1	.6	.8		
Ge		.9			.6	1	1	.6	.6
Fr		1			.8	1	1	.4	.4
UK		.2				.6	.4	1	1
Sp		.2				.6	.4	1	1

$\approx_n$	Cn	In	US	Ru	Jp	Ge	Fr	UK	Sp
Cn	1	1		.7		.4		1	1
In	1	1		.6		.4		.9	1
US			1						
Ru	.7	.6		1	.1	1		1	.9
Jp				.1	1	.4	.6		
Ge	.4	.4		1	.4	1		.8	.6
Fr					.6		1		
UK	1	.9		1		.8		1	1
Sp	1	1		.9		.6		1	1

Remark 6. The non-redundant basis T of a ranked table $\mathcal{D}$ of Theorem 5 can be efficiently computed by an algorithm with a polynomial time delay. We omit the presentation of the resulting algorithm due to space limitations and refer the reader to (Belohlavek & Vychodil, 2006b) for details.

EXAMPLE. We now present an example of a non-redundant basis of a ranked table. We consider a linear scale of 11 truth degrees 0 (falsity) $< 0.1 < 0.2 < \cdots < 1$ (full truth) equipped with Łukasiewicz connectives (Hájek, 1998) and globalization. Fig. 4 describes power consumption of selected countries. The attributes denote name of the county, mass of coal (megatons) produced for power purposes, electricity (MW) produced by air power-plants, electricity (10^3 MW) produced by water power-plants, electricity (10^{12} MW) produced by nuclear power-plants. For simplicity, we use names as tuples' identifiers of tuples instead of values of attributes.

Introducing similarity relations enables us to gain more information from the data. Let our similarities be given by Fig. 5. Our purpose is neither to study methods of specifying suitable similarities for particular data nor argue that our choice of similarities is "the best one"–this is a matter connected with particular problem domain (geography and economy, in this particular example) and should be left to experts in the areas.

If ranks of tuples are as given by the $\mathcal{D}(t)$-column of Fig. 4, then the table can be seen as a result of a query "select power consumption of countries with *very large populations*". Intuitively, one may expect the minimal basis of such a table would be smaller than the basis of the latter one because now several tuples (like Spain, France, . . .) have a low rank (the populations are rather small). Indeed, the minimal non-redundant basis given by the algorithm mentioned in Remark 6 is (after reduction of left-hand and right-hand sides of FDs) the following:

$$\begin{array}{ll}
\{c,{}^{0.8}\!/w\} \Rightarrow \{w\}, & \{{}^{0.1}\!/c\} \Rightarrow \{{}^{0.4}\!/c,{}^{0.4}\!/w\}, \\
\{{}^{0.9}\!/c,{}^{0.8}\!/w\} \Rightarrow \{{}^{0.9}\!/w\}, & \{{}^{0.6}\!/a\} \Rightarrow \{{}^{0.7}\!/c,{}^{0.8}\!/a,{}^{0.7}\!/w\}, \\
\{{}^{0.9}\!/c\} \Rightarrow \{a,n\}, & \{{}^{0.9}\!/n\} \Rightarrow \{n\}, \\
\{{}^{0.8}\!/c\} \Rightarrow \{{}^{0.8}\!/a,{}^{0.7}\!/w,{}^{0.8}\!/n\}, & \{{}^{0.5}\!/a\} \Rightarrow \{{}^{0.7}\!/n\}, \\
\{{}^{0.1}\!/c,{}^{0.9}\!/a\} \Rightarrow \{a\}, & \{{}^{0.1}\!/w\} \Rightarrow \{{}^{0.4}\!/c,{}^{0.4}\!/w\}, \\
\{{}^{0.5}\!/c\} \Rightarrow \{{}^{0.7}\!/c\}, & \{{}^{0.5}\!/n\} \Rightarrow \{{}^{0.5}\!/a,{}^{0.7}\!/n\}, \\
\{{}^{0.1}\!/c,{}^{0.5}\!/a\} \Rightarrow \{{}^{0.7}\!/c,{}^{0.8}\!/a,{}^{0.7}\!/w\}, & \{\} \Rightarrow \{{}^{0.4}\!/a,{}^{0.4}\!/n\}, \\
\{{}^{0.1}\!/c,{}^{0.5}\!/w\} \Rightarrow \{{}^{0.7}\!/c,{}^{0.8}\!/a,{}^{0.7}\!/w,{}^{0.7}\!/n\}. &
\end{array}$$

The basis can be seen as an additional information supplied along with the query result. Note that if Fig. 4 is considered as a classical one (no ranks, no similarities), its minimal basis consists of three (classical) FDs, namely $\{a\} \Rightarrow \{c,w,n\}$, $\{w\} \Rightarrow \{c,a,n\}$, and $\{n\} \Rightarrow \{c,a,w\}$. Thus, attributes a, w, and n are all keys of the table. Contrary to the previous case with similarities and ranks, the basis does not yield any non-trivial information.

OVERVIEW OF APPROACHES TO FUNCTIONAL DEPENDENCIES IN CODD'S MODEL WITH SIMILARITIES

As mentioned above, there are quite many papers on similarity extensions of Codd's relational model of data. A detailed overview is far beyond the scope of this chapter. To keep our discussion specific, we focus our attention to functional dependencies in Codd's model extended by similarities. In addition to presenting an overview of various approaches, we compare several of them with the approach presented in Section "Ranked tables over domains with similarities and their data dependencies". We omit proofs and refer a reader to (Belohlavek & Vychodil, 2006f).

(Raju & Majumdar, 1988) is perhaps the most influential paper on FDs over domains with similarities. Their extension of Codd's model is a particular case of ranked tables over domains with similarities from Section "Ranked tables over domains with similarities" in that they consider only $[0,1]$ as a structure of truth degrees and they do not consider any (truth function of) logical connective of implication. (Raju & Majumdar, 1988) is probably the first approach considering both ranks and similarities. However, the meaning of ranks is intuitively not very clear in (Raju & Majumdar, 1988). While we interpret a rank assigned to a tuple as a degree to which the tuple matches a similarity-based query (see Section "Ranked tables over domains with similarities"), Raju and Majumdar describe a rank as a degree to which a tuple belongs to a table. Later on, in Example 3.1, they say that a rank can be interpreted as a possibility measure or a measure of association of the items of a tuple.

Raju and Majumdar consider ordinary FDs in their model, i.e., consider $A \Rightarrow B$ where A and B are crisp sets, and consider a FD $A \Rightarrow B$ true in a ranked table $\mathcal{D}$ if for all tuples t_1, t_2 with $\mathcal{D}(t_1) > 0$ and $\mathcal{D}(t_2) > 0$ we have

$$\min_{y \in Y, A(y)=1} (t_1[y] \approx_y t_2(y)) \leq \min_{y \in Y, B(y)=1} (t_1[y] \approx_y t_2(y)). \qquad (13)$$

Consider now the relationship of (13) to $||A \Rightarrow B||_{\mathcal{D}}$ from Definition 3. We limit ourselves to the following points. First, Raju & Majumdar (1988) consider only "true" and "not true" for a given FD $A \Rightarrow B$. Thus, they disregard possible intermediate truth degrees to which $A \Rightarrow B$ may be true in $\mathcal{D}$. This may seem not natural in the context of domain similarities, because one wishes to have means to say that $A \Rightarrow B$ is "almost true", i.e., true in degree e.g. 0.9. Second, the expressive capability of FDs from (Raju & Majumdar, 1988) is smaller than that of our fuzzy FDs from Section "Ranked tables over domains with similarities". This is due to the restriction of A and B to be crisp sets. For instance, using FDs of Raju &

Majumdar (1988), it is not possible to describe a dependence “if similarity of t_1 and t_2 in y is at least 0.5 then similarity of t_1 and t_2 in y' is at least 0.8” (we omit details). Third, Raju & Majumdar (1988) do not make use of ranks $\mathcal{D}(t)$ in evaluation of validity of FDs in that it only matters whether or not $\mathcal{D}(t) > 0$. The concept $A \Rightarrow B$ of being true in $\mathcal{D}$ according to Raju & Majumdar (1988) is a particular case of the concept of $||A \Rightarrow B||_{\mathcal{D}}$ in the following sense (we omit proof):

Lemma 1. *For $L = [0,1]$, A,B crisp, * being identity, and arbitrary $\rightarrow$, denote for a given $\mathcal{D}$ by $\mathcal{D}'$ a ranked table defined by $\mathcal{D}'(t) = 1$ if $\mathcal{D}(t) > 0$ and $\mathcal{D}'(t) = 0$ otherwise. Then $A \Rightarrow B$ is true in $\mathcal{D}$ according to* (13) *iff $||A \Rightarrow B||_{\mathcal{D}'} = 1$ according to Definition 3.*

Fourth, also with other logical notions like that of semantic consequence, Raju & Majumdar (1988) consider these notions as bivalent (e.g., either $A \Rightarrow B$ follows from a set T of FDs or not).

Our last remark concerns Raju and Majumdar’s result on completeness of Armstrong’s axioms, see (Maier, 1983), w.r.t. to their semantics given by ranked tables with similarities. Raju and Majumdar essentially proved (although they presented their result in a bit different way) that any set $\mathcal{R}$ of deduction rules which is complete w.r.t. ordinary Codd’s model is also complete w.r.t. semantics given by ranked tables with similarities from (Raju & Majumdar, 1988). That is, a FD $A \Rightarrow B$ semantically follows from a set T of FDs (i.e., $A \Rightarrow B$ is true in each $\mathcal{D}$ such that any FD from T is true in $\mathcal{D}$) iff $A \Rightarrow B$ can be inferred from T using rules from $\mathcal{R}$. For this purpose, they elaborated quite a long proof in (Raju & Majumdar, 1988). Since the completeness holds for any system of ordinary Armstrong deduction rules (i.e., intermediate degrees in a sense do not matter in proofs), one might wonder whether it is possible to obtain the result by some simple reduction to the well-known completeness of the ordinary Codd’s model, see (Maier, 1983). This is, indeed, the case. Namely, the completeness result of Raju & Majumdar (1988) follows almost immediately from the following lemma (we omit proof):

Lemma 2. *A FD $A \Rightarrow B$ follows from a set T of FDs in the sense of Raju & Majumdar (1988) (semantics given by ranked tables with similarities) iff $A \Rightarrow B$ follows from T in the sense of ordinary Codd’s model (Maier, 1983).*

This way, we obtain a short proof which, moreover, provides us with an insight about the ordinary semantic consequence and that of Raju & Majumdar (1988). Note also that the completeness result of Raju & Majumdar (1988) can also be obtained as a consequence of Theorem 1 (we omit details here). At this point we stop our visit to (Raju & Majumdar, 1988).

Further approaches to FDs over domains with similarities The paper by Raju and Majumdar has been subject to several extensions. We now briefly comment on some of them, cf. (Belohlavek & Vychodil, 2006f). In all of the subsequent approaches, a FD is considered as a formula $A \Rightarrow B$ where A and B are crisp (i.e., $A \Rightarrow B$ is an ordinary FD) possibly with additional parameters, and FDs are being evaluated in data tables $\mathcal{D}$ over domains with similarities (without ranks). In addition to that, L is always confined to $[0,1]$. In (Cubero, & Vila, 1994), FDs are parameterized by $c_y \in [0,1]$ $(y \in Y)$. Values c_y are fixed and common to any FDs considered. A FD $A \Rightarrow B$ (denoted by the authors by $A \Rightarrow_{(\alpha,\beta)} B$ with $\alpha = (c_y)_{y\in A}$ and $\beta = (c_y)_{y\in B}$) is considered true in $\mathcal{D}$ whenever: if for each $y \in A$ we have $t_1[y] \approx_y t_2[y] \geq c_y$ then for each $y \in B$ we have $t_1[y] \approx_y t_2[y] \geq c_y$. One can see that if we define fuzzy sets A_c and B_c by $A_c(y) = c_y$ for $y \in A$ and $A_c(y) = 0$ for $y \notin A$ (and the same for B_c), we have (proof omitted):

Lemma 3. *For * being globalization, $L = [0,1]$, and arbitrary $\rightarrow$, $A \Rightarrow B$ is true in $\mathcal{D}$ according to Cubero, & Vila (1994) iff $||A_c \Rightarrow B_c||_{\mathcal{D}} = 1$, cf. (12).*

Therefore, (Cubero, & Vila, 1994) results as a particular instance of the approach from Section “Ranked tables over domains with similarities and their data dependencies”. Moreover, the approach from Section “Ranked tables over domains with similarities and their data dependencies” does not require fixed thresholds c_y.

(Ben Yahia, Ounalli, & Jaoua, 1999) is one of few papers which consider degrees of validity of FDs in tables $\mathcal{D}$ with similarity relations. Tables $\mathcal{D}$ do not have ranks and FDs are ordinary FDs $A \Rightarrow B$ with both A and B ordinary sets of attributes. In (Ben Yahia, Ounalli, & Jaoua, 1999), a FD $A \Rightarrow B$ is considered true to degree b in $\mathcal{D}$ if

$$b = \min_{t_1,t_2}(\min(1, 1 - (t_1(A) \approx_{\mathcal{D}} t_2(A) + (t_1(B) \approx_{\mathcal{D}} t_2(B)))$$

and if for all tuples t_1, t_2 we have

$$\min(1, 1 - (t_1(A) \approx_{\mathcal{D}} t_2(A) + (t_1(B) \approx_{\mathcal{D}} t_2(B)) \geq \theta$$

where $\theta \in [0,1]$ is an additional parameter. The basic relationship to our approach is the following:

Lemma 4. *For * being identity, $L = [0,1]$, and Łukasiewicz $\rightarrow$, $A \Rightarrow B$ is true in degree b in $\mathcal{D}$ according to Ben Yahia, Ounalli, & Jaoua (1999) iff $b = ||A \Rightarrow B||_{\mathcal{D}}$ and $b \geq \theta$, cf.* (12).

A similar approach is adopted in (Bhuniya & Niyogi, 1993). In (Chen, Kerre, & Vandenbulcke, 1994), a FD $A \Rightarrow_\gamma B$ (with $\gamma \in [0,1]$ a parameter) is considered true in $\mathcal{D}$ if $A \Rightarrow B$ is true in $\mathcal{D}$ in the ordinary sense and if

$$t_1(A) \approx_{\mathcal{D}} t_2(A) \rightarrow t_1(B) \approx_{\mathcal{D}} t_2(B) \geq \gamma$$

where $\rightarrow$ is Gödel implication, cf. Section "Preliminaries". The following relationship to our model is almost obvious:

Lemma 5. *For * being identity, $L = [0,1]$, and Gödel $\rightarrow$, $A \Rightarrow_\gamma B$ is true in $\mathcal{D}$ according to Chen, Kerre, & Vandenbulcke (1994) iff $||A \Rightarrow B||_{\mathcal{D}} \geq \gamma$, cf.* (12), *and if $A \Rightarrow B$ is true in $\mathcal{D}$ as an ordinary FD.*

The above example illustrate that in some of the approaches which can be found in the literature we can find attempts to capture similarity thresholds in FDs as well as attempts which try to capture approximate validity of FDs in tables with similarities on domains. These attempts are very particular cases of the approach from Section "Ranked tables over domains with similarities and their data dependencies". From our analysis, we can see the following advantages of the approach from Section "Ranked tables over domains with similarities and their data dependencies":

- The approach from Section "Ranked tables over domains with similarities and their data dependencies" makes it possible to capture both similarity thresholds and approximate validity of FDs at once.
- The approach is versatile and has greater expressive capability. First, similarity thresholds need not be fixed. Second, a whole spectrum of various logical connectives can be used, rather than a fixed connective, such as Łukasiewicz implication in (Ben Yahia, Ounalli, & Jaoua, 1999).
- The approach naturally includes ranks $\mathcal{D}(t)$.
- Most importantly however, the approach from Section "Ranked tables over domains with similarities and their data dependencies" is conceptually clean: the logical formula behind the approach is the same as the logical formula in the case of ordinary FDs.

Further papers on FDs in similarity extension of Codd's model include, e.g., (Bosc, Dubois, & Prade, 1994; Intan & Mukaidono, 2004, Liu, 1997; Shenoi & Melton, 1992; Tyagi, Sharfuddin, Dutta, & Tayal, 2005; Wang, Tsai, & Hong, 2000). These papers, and papers related to these papers, are discussed in (Belohlavek & Vychodil, 2006f) and will be subject of future research.

Prade and Testemale and related approaches (Prade & Testemale, 1984) is a seminal paper on another extension of the relational model from the point of view of fuzzy logic. Due to lack of space we only briefly comment on it. The main idea of this extension consists in allowing fuzzy sets A_y in domains D_y as members of the tuples. FDs in this model are based on similarity relations between fuzzy sets A_y and A'_y. An important observation here is that one can consider the FDs in this model as a particular case of FDs in (ranked) tables over domains with similarities. Namely, the elements of the domains are fuzzy sets A_y and the similarities on the domains are the above-mentioned similarity relations between fuzzy sets. Then, one can employ results on FDs of tables over domains with similarities to the model of Prade & Testemale (1984) and to related models.

CONCLUSIONS AND RESEARCH TOPICS

We have demonstrated some of the benefits of developing fuzzy logic extensions of Codd's relational model according to the principles of fuzzy logic in narrow sense by presenting an example of such an extension in Section "Ranked tables over domains with similarities and their data dependencies". Furthermore, we compared some of the influential approaches from the literature with the presented extension and derived some new observations and results. The main conclusion is that not only are several of the approaches proposed in the literature particular instances of the extension from Section "Ranked tables over domains with similarities and their data dependencies" but, more importantly, our extension, based on the principles of fuzzy logic, is transparent and tractable from both theoretical and algorithmic point of view. The following list includes topics for future research in data dependencies in Codd's model of data extended by similarity relations on domains.

(1) A comprehensive comparison of approaches to data dependencies in Codd's model with similarities. Emphasis needs to be put on conceptual issues. Namely, several papers address the same conceptual issues in different ways. However, the differences are only technical in nature, i.e., not essential. As we have seen, it is often the case that seemingly different approaches are particular instances of a more general approach which is more transparent. A useful output of such a comparison would be a list of essential conceptual issues which were identified in previous papers.

(2) A systematic study of all kinds of dependencies which proved to be useful in the ordinary Codd's model of data. Up to now, studies of other kinds of dependencies than functional dependencies in Codd's model with similarities are exceptions.

(3) Data dependencies in Codd's model with similarities have a data mining appeal. Several approaches to extracting functional dependencies from data have already appeared, see, e.g., (Belohlavek & Vychodil, 2006b; Chen, Kerre, & Vandenbulcke, 1994; Manilla & Räiha, 1994; Wand, Tsai, & Hong, 2000; Wei & Chen, 2004). However, a state-of-the-art survey is not available and the current status in these problems is not clear. A further research in this area is needed.

(4) The above topics are a part of a general goal of developing solid foundations of relational model of data with similarities on domains. As repeated several times in this paper, a clear connection to fuzzy logic in narrow sense is necessary to have a conceptually clear and consistent framework. The advanced state of the art of fuzzy logic in narrow sense provides us with a solid bunch of notions and results which can be utilized. Relational algebra for relational model with similarities on domains is a particular topic in this respect to which attention has been paid in the past. Papers on this topic include (Belohlavek & Vychodil, 2006c; Buckles, Petri, Sachar, 1989; Penzo, 2005; Prade & Testemale, 1984; Takahashi, 1993). However, the available results still need to be considered as preliminary attempts rather than a definite solution. It is important to note in this connection that several conceptually related papers appeared recently in database community, see, e.g., (Fagin, 1999, 2002; Illyas, Aref, & Elmagarmid, 2004; Li, Chang, Illyas, & Song, 2005). Moreover, management of uncertainty and imprecision in database systems is considered an important goal by senior database researchers (Abiteboul, Agrawal, Bernstein, Carey, Ceri, Croft, DeWitt, *et al.*, 2005). These are important signals supporting the general message of this chapter:

Many particular approaches to management of imprecision and management of similarity exist in the database literature. Management of uncertainty and imprecision is recognized as an utmost important problem by database community. In spite of a long-term effort, solid comprehensive foundations for relational model of data with similarities on domains and for data dependencies in this model in particular are not available. One of the main reasons is a lack of conceptual clarity and a lack of versatility of the proposed solutions. At this point of time, however, the state of the art in fuzzy logic in narrow sense provides firm foundations for the development of relational model with similarities on domains. Such development could substantially improve the way humans access information using computers.

REFERENCES

Abiteboul S., Agrawal R., Bernstein P., Carey M., Ceri S., Croft B., DeWitt D., *et al.* (2005). The Lowell database research self-assessment. *Comm. ACM 48*(5), 111–118.

Armstrong W. W. (1974). Dependency structures in data base relationships. *IFIP Congress*, International Federation of Information Processing, Stockholm, Sweden, pp. 580–583.

Belohlavek R. (2002). *Fuzzy Relational Systems: Foundations and Principles*. Kluwer, Academic/Plenum Publishers, New York.

Belohlavek R., & Vychodil V. (2005). Functional dependencies of data tables over domains with similarity relations. In *Proc. IICAI 2005*, Indian International Conference on Artificial Intelligence, Pune, India, ISBN 0–9727412–1–6, pp. 2486–2504.

Belohlavek R., & Vychodil V. (2006a). Attribute implications in a fuzzy setting. In Missaoui R., Schmid J. (Eds.), *ICFCA 2006*, International Conference on Formal Concept Analysis, *LNAI 3874*, ISBN 3–540–32203–5, pp. 45–60.

Belohlavek R., & Vychodil V. (2006b). Data tables with similarity relations: functional dependencies, complete rules and non-redundant bases. In Lee M. L., Tan K. L., & Wuwongse V. (Eds.), *DASFAA 2006*, Database Systems for Advanced Applications, *LNCS 3882*, ISBN 3–540–33337–1, pp. 644–658.

Belohlavek R., & Vychodil V. (2006c). Relational Model of Data over Domains with Similarities: An Extension for Similarity Queries and Knowledge Extraction. In *IEEE IRI 2006*, Information Reuse and Integration, Hawaii, USA, ISBN 0–7803–9788–6, pp. 207–213.

Belohlavek R., & Vychodil V. (2006d). Fuzzy attribute logic over complete residuated lattices. *J. Experimental and Theoretical Artificial Intelligence 18*, 471–480.

Belohlavek R., & Vychodil V. (2006e). Pavelka-style fuzzy logic for attribute implications. In *Proc. JCIS 2006, Joint Conference on Information Sciences, Kaohsiung, Taiwan, ROC,* pp. 1156–1159.

Belohlavek R., & Vychodil V. (2006f). Codd's relational model of data and fuzzy logic: comparisons, observations, and some new results. In *Proc. CIMCA 2006*, Computational Intelligence for Modelling, Control and Automation, Sydney, Australia, ISBN 0–7695–2731–0, 6 pages.

Bhuniya B., & Niyogi P. (1993). Lossless join property in fuzzy relational databases. *Data and Knowledge Engineering 11*(2), 109–124.

Ben Yahia S., Ounalli H., & Jaoua A. (1999). An extension of classical functional dependency: dynamic fuzzy functional dependency. *Information Sciences 119*, 219–234.

Bosc P., Dubois D., & Prade H. (1994). Fuzzy functional dependencies. An overview and a critical discussion. In *FUZZ-IEEE '94*, IEEE Int. Conference on Fuzzy Systems, pp. 325–330.

Buckles B. P., & Petry F. E. (1982). A fuzzy representation of data for relational databases. *Fuzzy Sets and Systems 7*, 213–226.

Buckles B. P., Petry F. E., & Sachar H. (1989). A domain calculus for fuzzy relational databases. *Fuzzy Sets and Systems 29*, 327–340.

Chen G., Kerre E. E., & Vandenbulcke J. (1994). A computational algorithm for the FFD transitive closure and a complete axiomatization of fuzzy functional dependence (FFD). *Int. J. Intelligent Systems*, *9*, 421–439.

Cubero J. C., & Vila M. A. (1994). A new definition of fuzzy functional dependency in fuzzy relational datatabses. *Int. J. Intelligent Systems 9*, 441–448.

Date C. J. (2000). *Database Relational Model: A Retrospective Review and Analysis.* Addison Wesley.

Dey D., & Sarkar S. (1996). A probabilistic relational model and algebra. *ACM Transactions on Database Systems 21*, 339–369.

Fagin R. (1999). Combining fuzzy information from multiple systems. *J. Computer and System Sciences 58*, 83–99.

Fagin R (2002). Combining fuzzy information: an overview. *ACM SIGMOD Record 31*(2), 109–118.

Fuhr N., & Rölleke T. (1997). A probabilistic relational algebra for the integration of information retrieval and database systems. *ACM Transactions on Information Systems 15*, 32–66.

Galindo J., Urrutia, A. & Piattini M. (2006). *Fuzzy Databases: Modeling, Design and Implementation.* Idea Group Publishing, Hershey, PA.

Ganter B., & Wille R. (1999). *Formal Concept Analysis. Mathematical Foundations.* Springer, Berlin.

Gerla G. (2001). *Fuzzy Logic. Mathematical Tools for Approximate Reasoning.* Kluwer, Dordrecht.

Guigues J. L., & Duquenne V. (1986). Familles minimales d'implications informatives resultant d'un tableau de données binaires. *Math. Sci. Humaines 95*, 5–18.

Hajek P. (1998). *Metamathematics of Fuzzy Logic.* Kluwer, Dordrecht.

Hajek P. (2001). On very true. *Fuzzy Sets and Systems 124*, 329–333.

Illyas I. F., Aref W. G., & Elmagarmid A. K. (2004). Supporting top-k join queries in relational databases. *The VLDB Journal 13*, 207–221.

Intan R., & Mukaidono M. (2004). Fuzzy conditional probability relations and their applications in fuzzy information systems. *Knowledge and Information Systems 6*, 345–365.

Klir G. J., & Yuan B. (1995). *Fuzzy Sets and Fuzzy Logic. Theory and Applications.* Prentice Hall.

Li C., Chang K. C.-C., Ilyas I. F., & Song S. (2005). RanSQL: Query Algebra and Optimization for Relational top-k queries. In *ACM SIGMOD 2005*, 131–142.

Liu W.-Y. (1997). Fuzzy data dependencies and implication of fuzzy data dependencies. *Fuzzy Sets and Systems 92*, 341–348.

Maier D. (1983). *The Theory of Relational Databases.* Computer Science Press, Rockville.

Manilla H., & Räiha K. J. (1994). Algorithms for inferring functional dependencies from relations. *Data & Knowledge Engineering 12*, 83–99.

Medina, J.M., Pons, O. & Vila, M.A. (1994). GEFRED. A Generalized Model of Fuzzy Relational Databases. *Information Sciences 76* (1-2), 87–109.

Novák V., Perfilieva I., & Močkoř J. (1999). *Mathematical Principles of Fuzzy Logic.* Kluwer, Dordrecht.

Pavelka J. (1979). On fuzzy logic I, II, III. *Zeitschrift für Mathematische Logik Grundlagen der Mathematik 25*, 45–52, 119–134, 447–464.

Penzo W. (2005). Rewriting rules to permeate complex similarity and fuzzy queries within a relational database system. *IEEE Transactions on Knowledge and Data Engineering 17*, 255–270.

Petry F. (1996). *Fuzzy Databases: Principles and Applications.* Kluwer Academic.

Prade H., & Testemale C. (1984). Generalizing database relational algebra for the treatment of incomplete or uncertain information and vague queries. *Information Sciences 34*, 115–143.

Prade H., & Testemale C. (1988). Fuzzy relational databases: representational issues and reduction using similarity measures. *J. American Society for Information Science 38*(20), 118–126.

Raju K. V. S. V. N., & Majumdar A. K. (1988). Fuzzy functional dependencies and lossless join decomposition of fuzzy relational database systems. *ACM Transactions on Database Systems 13*(2), 129–166.

Sharma A. K., Goswami A., & Gupta D. K.. (2004). Fuzzy inclusion dependencies in fuzzy relational databases. In *ITCC'04*, Int. Conf. on Information Technology: Coding and Computing, Vol. 2, pp. 507–510.

Shenoi S, & Melton A. (1992). Functional dependencies and normal forms in the fuzzy relational database model. *Information Sciences 100*, 1–28.

Takahashi Y. (1993). Fuzzy database query languages and their relational completeness theorem. *IEEE Transactions on Knowledge and Data Engineering 5*, 122–125.

Takeuti G., & Titani S. (1987). Globalization of intuitionistic set theory. *Annals of Pure and Applied Logic*

33, 195–211.

Tyagi B. K., Sharfuddin A., Dutta R. N., & Tayal D. K. (2005). A complete axiomatization of fuzzy functional dependencies using fuzzy function. *Fuzzy Sets and Systems 151*, 363–379.

Ullman D. D. (1988). *Principles of Database and Knowledge-Base Systems.* Computer Science Press, New York.

Wang S.-L., Tsai J.-S., & Hong T.-P. (2000). Mining Functional Dependencies from Fuzzy Relational Databases. In *ACM SAC 2000*, Symposium on Advanced Computing, pp. 490–493.

Wei Q., & Chen G. (2004). Efficient discovery of functional dependencies with degrees of satisfaction. *Int. J. on Intelligent Systems 19*, 1089–1110.

Key Terms and Their Definitions

Codd's Relational Model of Data: A theoretical model of data representation and manipulation due to Edgar F. Codd (1960s, 1970s). Data is conceived as represented by tables in Codd's model. A formal counterpart of a table is that of a relation. Data manipulation corresponds to performing operations with relations. Codd's model relies on first-order logic and a mathematical concept of a relation. Codd's relational model is the theoretical backbone of relational databases.

Functional Dependency: Functional dependency is a formula $A \Rightarrow B$ where A and B are collections of attributes. $A \Rightarrow B$ being true in a table means that every two rows of a table which have the same values on attributes from A have the same values on attributes from B. Functional dependencies play important role in design of relational databases.

Domain With Similarity: Domain is a set of all possible values an attribute may take. For instance, a domain of attribute "age" is a set $\{0,1,2,\ldots,150\}$. A domain with similarity is a domain equipped with a particular binary fuzzy relation on it, called a similarity relation, i.e. with a function assigning to every two elements of the domain a degree to which the two values are similar.

Armstrong Rules: Armstrong rules are deduction rules for reasoning with functional dependencies. Usually, by Armstrong rules we mean a collection of rules which are syntactico-semantically complete. That is, a functional dependency $A \Rightarrow B$ semantically follows from a set T of functional dependencies iff $A \Rightarrow B$ can be obtained from T using Armstrong rules.

Structure of Truth Degrees: A set of truth degrees such as $[0,1]$ equipped with truth functions of logical connectives. For instance, for the connective of implication, one can use $a \rightarrow b = \min(1, 1-a+b)$ ()Lukasiewicz implication), or $a \rightarrow b = 1$ for $a \leq b$ and $a \rightarrow b = b/a$ for $a > b$ (Goguen implication). There are many choices of truth functions of logical connectives. However, the chosen collection of connectives should obey reasonable properties such as the adjointness property which is required to be satisfied by truth functions of conjunction and implication.