Polska Akademia Nauk • Instytut Badań Systemowych

# AUTOMATYKA
# STEROWANIE
# ZARZĄDZANIE

Książka jubileuszowa
z okazji
70-lecia urodzin

PROFESORA KAZIMIERZA MAŃCZAKA

pod redakcją
Jakuba Gutenbauma

Polska Akademia Nauk • Instytut Badań Systemowych

# AUTOMATYKA
# STEROWANIE
# ZARZĄDZANIE

**Książka jubileuszowa**
**z okazji**
**70-lecia urodzin**

**PROFESORA KAZIMIERZA MAŃCZAKA**

**pod redakcją**
**Jakuba Gutenbauma**

Warszawa 2002

Książka jubileuszowa z okazji
70-lecia urodzin
Profesora Kazimierza MAŃCZAKA

Redaktor
prof. dr hab. inż. Jakub Gutenbaum



**Warszawa 2002**

**ISBN 83-85847-78-2**

**Wydawca: Instytut Badań Systemowych PAN**
**ul. Newelska 6 01-447 Warszawa**
**http://www.ibspan.waw.pl**

Opracowanie składopisu: Anna Gostyńska, Jadwiga Hartman

Druk: KOMO-GRAF, Warszawa
nakład 200 egz., 34 ark. wyd., 31 ark. druk.

# FUZZY LOGIC AND RELATIONAL DATABASES: SELECTED APPLICATIONS

***Sławomir Zadrożny***
*Systems Research Institute, Polish Academy of Sciences*

***Abstract.*** *The applicability of the selected fuzzy logic related concepts for the purposes of data representation and manipulation in relational databases is considered. The concepts discussed include fuzzy logical connectives for data querying and possibility distributions for data representation.*

***Keywords:*** *relational databases, fuzzy logic, possibility distributions, fuzzy logical connectives.*

## 1. Introduction

Among the main issues of any data model there are a data representation scheme and a data manipulation formalism. In case of the former, the main concern is the logical data representation, i.e., how the data are arranged from the user perspective, rather than how they are physically represented on some storage devices. For the latter, the data retrieval is the most important operation. Both issues are successfully addressed within the relational data model (Codd 1970). The consistent use of the mathematical concept of relation and the relational algebra for data representation and manipulation proved to be adequate and made the model widely accepted. However, there are still some issues that require further research. One of such issues is dealing with imperfect information. In many typical applications it may be assumed that the user possesses complete knowledge about all relevant facts and relationships in the modelled segment of the real world. Thus, in such a context, for both representation and retrieval of data crisp, precise values are sufficient. However, such an ideal situation cannot be assumed for all the areas of potential application of the database supported systems. Several approaches have been proposed to deal with this problem. None of them obtained a widespread acceptance so far. Nevertheless, this seems to be an important and current problem. The problem of representation and processing of imperfect information may be seen as an important element of *human consistency* of any software system. In a real life situation we usually possess only a partial, incomplete information. Moreover, we are accustomed to express our requirements in the natural language which is inherently vague, imprecise. Thus, we expect the software systems will help us to solve our problems and will be able to accept incomplete parameters as well as imprecisely stated goals, preferences etc. Fuzzy logic has been conceived exactly to the aim of dealing with such questions. In this paper we will briefly present selected concepts of fuzzy logic that are applicable for data representation and retrieval within the general framework of the relational data model.

The essence of the relational model (Date 1995) is providing the user with a unified logical perspective of a database as a collection of relations. Basically, all information about the modelled part of the world is represented in the form of *tables* being a more down-to-earth metaphor for the mathematical relations. We can make a more precise distinction assuming the following terminology. A relation is, as usually, a subset of the Cartesian product of certain sets (called here *domains*) $D_1,\ldots,D_n$. Thus, a relation may be characterized by a name, $R$, and a set of *attributes*, $A_1,\ldots,A_n$, where each attribute $A_i$ may take on a value from the corresponding domain $D_i$. The set of pairs $(A_i, D_i)$ is referred to as the *scheme* of the relation $R$. A table is a data structure suitable to host the tuples of a relation of a certain scheme. Thus, the columns of the table correspond to the attributes of the relation and the rows of the table correspond to the tuples of this relation. The value of an attribute $A$ in tuple $t$ will be denoted with $t[A]$.

In the next section we discuss the querying of the regular (*crisp*) relational databases. It is shown how the fuzzy logic based concepts make it possible to go beyond the classical Boolean scheme of querying and grasp the imprecision typical for the natural language related queries. Section 3 deals with the representation of the incomplete information in a database using the concept of possibility distribution. The question of the querying of such a database is also discussed.

## 2. Fuzzy logical connectives in queries to a crisp database

The retrieval of relevant data from a database is by far the most important operation. In the framework of the relational data model there are two basic formalisms to specify a query, i.e., what data is sought for: the *relational algebra* and the *first order predicate calculus*. The result of a query is always a relation containing the required data. The query in the first formalism is a sequence of the operations of the relational algebra such as selection (restriction), projection, join, union or intersection. When this sequence of operations is executed against a database one obtains the required data. The second formalism is of more declarative character and employs the predicate calculus formula to describe the relevant data as the data satisfying the formula. How it is to be found in a database is a matter of the database management system.

Whatever querying formalism is used always a part of the query may be a set of conditions (criteria) specifying which rows will be *selected* to be included in an answer to the query. Thus, it is interesting to study the retrieval process from the perspective where a query is meant to define by means of these conditions a prototype of data to be retrieved. Then, during the retrieval process for every row a *matching degree* of its content and the prototype is calculated. In classical crisp approach this matching degree is binary: a row matches the prototype or not. In the practical situations, the description of the prototype may be imprecise what in a natural way leads to a partial matching degree. Such an approach has several advantages. First, the user is not forced to specify the criteria unnecessarily precisely. For example, while looking for a house it may be much more natural to require it to be "cheap" or "large" instead of specifying exactly the acceptable price interval or the size of the land area. Moreover, there may be houses almost meeting our criteria that would be excluded by crisp Boolean conditions and will be included in the answer to a fuzzy query with just reduced matching degree. Next, the matching degree makes it possible to order the results of the query from best to least satisfying our criteria. These may be very convenient when the result set is large. These benefits of fuzzy queries are widely advocated (Bosc, Pivert 1992, 1995 and Kacprzyk, Zadrożny 1995, 1997). In what follows we will be concerned with an important issue of the fuzzy queries processing: the aggregation of the partial matching degrees. In case of the crisp, Boolean queries the aggregation of the partial matching degrees is naturally carried on using classical logical connectives. Thus, we start with a brief review of their counterparts proposed within the fuzzy logic. Then, we discuss various types of fuzzy queries proposed in the literature. We focus on the sophisticated aggregation schemes involving fuzzy linguistic quantifiers, importance weights, hierarchies of the partial conditions.

There are several possible generalizations of the classical logical connectives within the fuzzy logic. Among them, the most commonly adopted are the following, originally introduced by Zadeh:

$$x \wedge y = \min(x, y) \tag{1}$$

$$x \vee y = \max(x, y) \tag{2}$$

$$\neg x = 1 - x \tag{3}$$

for conjunction, disjunction and negation connectives, respectively.

Several fuzzy implication operators have been proposed in the literature. The most commonly used are:

- Kleene-Dienes: $I(x,y) = \max(1 - x, y)$ (4)
- Łukasiewicz: $I(x,y) = \min(1, 1-x+y)$ (5)
- Gödel: $I(x, y) = \begin{cases} 1 & \text{if } x \leq y \\ y & \text{otherwise} \end{cases}$ (6)
- Goguen: $I(x,y) = \min(y/x, 1)$ (7)

Zadeh (1983) introduced two types of *linguistically quantified propositions*:

$Q\ X\text{'s}$ are $C$'s (type I); (8)

$QB\ X\text{'s}$ are $C$'s (type II), (9)

where $Q$ is a linguistic quantifier, and $C$ and $B$ are fuzzy sets in the universe $U$. Fuzzy linguistic quantifiers are represented by fuzzy sets defined in an appropriate universe. The *absolute* linguistic quantifiers such as "approximately 3", "several" are represented as fuzzy subsets on the positive real numbers $R^+$ domain; *proportional* linguistic quantifiers such as "most", "almost all", etc. are represented by fuzzy subsets on the interval [0,1]. Zadeh proposed the interpretation for proportional linguistic quantifiers such that the truth degree $T$ of the proposition (8) is computed using the following formula:

$$T = \mu_Q\left(\frac{\text{card}(C)}{\text{card}(X)}\right) = \mu_Q\left(\frac{\sum_i \mu_C(x_i)}{n}\right) \tag{10}$$

where $\mu_Q$ is the membership function of the quantifier $Q$ and $n$ is the cardinality of the universe $U$. For propositions of the type (9) we have:

$$T = \mu_Q\left(\frac{\mathrm{card}(C \cap B)}{\mathrm{card}(B)}\right) = \mu_Q\left(\frac{\sum_i (\mu_C(x_i) \wedge \mu_B(x_i))}{\sum_i \mu_B(x_i)}\right) \tag{11}$$

Tahani (1977) was first to propose the use of the elements of fuzzy logic to enhance the flexibility of crisp database queries. He proposed a formal approach and architecture to deal with simple fuzzy queries for crisp relational databases. The idea may be best illustrated on an example of a query:

> "Find the names and department numbers of employees who are *young* and have a *high* salary OR those who are *young* and have *low* commission".

Thus, Tahani proposed to use in query's condition vague terms typical for natural language. The semantics of these vague terms is provided by appropriate fuzzy sets. Then, the main question is how the matching degree for the query is computed. For that purpose Tahani defines the matching function $\gamma$. For a tuple $t$ and a simple query $q$ of type $A = F$, where $A$ is an attribute (e.g., "age") and $F$ is a vague (fuzzy) term (e.g., "young"), the value of the function $\gamma$ is:

$$\gamma(q,t) = \mu_F(u) \tag{12}$$

where $u$ is $A(t)$, i.e. the value of tuple $t$ for the attribute $A$. The matching function $\gamma$ for more complex queries involving logical connectives is as follows:

$$\gamma(p \text{ AND } q, t) = \min(\gamma(p, t), \gamma(q, t)) \tag{13}$$

$$\gamma(p \text{ OR } q, t) = \max(\gamma(p, t), \gamma(q, t)) \tag{14}$$

$$\gamma(\neg q, t) = 1 - \gamma(q, t) \tag{15}$$

where $p$, $q$ are queries (partial conditions). Thus, the logical connectives in queries are interpreted as the original Zadeh's fuzzy connectives.

Kacprzyk and Ziółkowski (1986) were among the first to propose the aggregation of partial conditions of a query to be guided by a linguistic quantifier. Thus, they proposed to extend the querying language so that the query's condition would be expressed as:

$$q = Q \textit{ out of } \{p_1, \ldots, p_k\} \tag{16}$$

where $Q$ is a linguistic (fuzzy) quantifier and $p_i$ are partial conditions (queries) to be aggregated. Thus, the overall matching degree is computed using the semantics of the quantifier involved and (8) becomes

$$\gamma(q,t) = \mu_Q\left(\frac{\sum_i \gamma(p_i,t)}{n}\right) \tag{17}$$

where $\gamma(q,t)$ is, as in (12), the matching degree of the overall query. In Kacprzyk and Ziółkowski (1986) the original Zadeh's approach, leading to (17), has been adopted and later (Kacprzyk and Zadrożny 1997) the OWA operators (Yager 1988) were employed to model the linguistic quantifier. Both type I and type II of linguistically quantified propositions were studied in this context. In the latter case the query of (16) may be extended to:

$$q = Q \text{ important } \mathit{out\ of} \{p_1, \ldots, p_k\} \tag{18}$$

where the importance is represented by a fuzzy set of partial conditions in such a way that the value of the membership function of given $p_i$ is equal to its importance weight.

Another scheme for the aggregation of fuzzy conditions of varying importance has been studied by Bosc and Pivert (1993). They proposed a fuzzy operator for the hierarchical aggregation of fuzzy conditions, which starts from the concept of hierarchical aggregation introduced by Lacroix and Lavency (1987) for crisp conditions. Lacroix and Lavency proposed to extend the concept of classical crisp queries in the following way. A query $q$ has two parts: a selection part, $S$, and a set of crisp conditions, $PRF$, called preferences. The semantics of this query is the following: select the tuples satisfying $S$ and rank them according to the $PRF$. More precisely, if there are no tuples satisfying condition $S$ then the answer to the query is empty. Otherwise, the answer comprises the tuples that verify $S$ and at the same time best satisfy $PRF$. In the latter case, various assumptions on the interrelation of the conditions belonging to the $PRF$ may be made. Two cases are considered: (1) the conditions are equally important, (2) there is a (linear) hierarchy of conditions - those higher in hierarchy are more important. Thus, in the second case we have the importance of conditions imposed not by the numerical weights, but by their position in the hierarchy. The ranking of the tuples depends on what assumption is made: (1) or (2). In the first case, the count of the conditions in $PRF$ that are satisfied by a tuple is taken into account. In the second case, the lexicographic ordering of the tuples according to their fulfilment of particular conditions belonging to $PRF$ (taken in order imposed by the hierarchy) is employed. Bosc and Pivert (1993) proposed a fuzzy operator $\gamma$ to model the hierarchical aggregation described above, in which the contribution of a condition $p_j \in PRF$ to the overall matching degree is less or equal than the contribution of conditions higher in the hierarchy. Let us assume that the conditions are ordered according to the hierarchy, i.e., if $i<j$ then $p_i$ is higher in the hierarchy (is more important) than $p_j$. The fuzzy operator proposed is defined as a combination of two operators. The first, denoted with $\eta$ limits the contribution of the condition $p_j$ relative to the contributions of all preceding conditions $p_i$ ($i<j$), while the second combines all contributions to obtain the final value for the aggregation of the fuzzy conditions,

$$\gamma(q,t)=\frac{\sum_{i=1}^{n}\eta(p_i,t)}{n} \tag{19}$$

where $\eta(p_i,t)=\min_{(j\leq i)}(\gamma(p_j,t))$ and $\gamma(p_j,t)$, as in (12), denotes the matching degree of partial condition $p_j$ and tuple $t$.

Bosc and Pivert adopt here a different interpretation of hierarchy of conditions than originally assumed by Lacroix and Lavency. Namely, in the latter case, if no tuple satisfies a condition from, e.g., $k$-th level of the hierarchy then the conditions of the lower levels do play a role in ranking the tuples. In the former approach, all these lower levels are neglected.

Dubois and Prade (1997) studied the question of conditions $p_i$ with varying degrees of importance forming together a compound condition $q$ via the conjunction. The first model considers some importance (weight) $w_i$ for each elementary condition $p_i$ and the matching degree of the weighted condition $p_i$ against a tuple $t$ is given by the following equation:

$$\gamma(p_i^*,t)=\max(\gamma(p_i,t),1-w_i)\,, \tag{20}$$

where $p_i^*$ denotes the condition $p_i$ with an importance associated to it. Then, the matching degree of the condition $q$ is calculated using standard min operator:

$$\gamma(p,t)=\min_i \gamma(p_i^*,t)=\min_i \max(\gamma(p_i,t),1-w_i) \tag{21}$$

Hence, when the importance is minimal ($w_i$ = 0), the condition $p_i$ is not considered in the evaluation. On the other hand, with $w_i$ = 1, the evaluation of condition $p_i$ highly influences the evaluation of the overall condition $q$. This model has been refined (Dubois, Prade 1997) to deal with a variable importance $w_i$ – depending on the matching degree of the associated elementary condition. For example, in a specific context it may be useful to assume $w_i$ constant for relatively high satisfaction of the elementary condition, but the extremely low satisfaction should be stronger reflected in the overall matching degree by automatically increased $w_i$. For instance, when we look for a car and we prefer one having a moderate price, but it is not our primary criterion (condition), then we assume the importance weight for it smaller than 1.0. However, if a particular car has a very high price, the price criterion becomes more important ($w_i = 1$) in order to reject that car.

The second model, originally proposed by Yager in 1984 (Dubois, Prade 1997) considers a threshold $\theta_i$ for each elementary condition $p_i$. If the condition $p_i$ is satisfied to a degree above the threshold $\theta_i$, that is, $\gamma(p_i,t) \geq \theta_i$, then the resulting partial matching degree becomes 1, i.e., $\gamma(p_i^*,t) = 1$. On the other hand, if the threshold is not reached, i.e., $\gamma(p_i,t) < \theta_i$, we may consider two ways for the evaluation of the condition: (AA) $\gamma(p_i^*,t) = \gamma(p_i,t)$ or (B) $\gamma(p_i^*,t) = \dfrac{\gamma(p_i,t)}{\theta_i}$. It turns out, that both ways may be expressed with a formula:

$$\gamma(q,t) = \min_i \gamma(p_i^*,t) = \min_i \theta_i \rightarrow \gamma(p_i,t) \tag{22}$$

where $\rightarrow$ is the implication logical operator. Then, using the Gödel implication (6) and the Goguen implication (7) we obtain (A) and (B), respectively. Note that the first model of importance, proposed by Dubois and Prade and formalized by the equation (20), is also covered by the general formula of (22) when the Kleene-Dienes implication (4) is assumed.

Still another model of importance applicable for the aggregation of partial matching has been proposed by Dubois and Prade (1997), in which they used conditional requirements $p_i \rightarrow p_j$ to provide an interpretation for the hierarchical aggregation of fuzzy predicates. The authors consider similar context to that of the paper by Lacroix and Lavency (1987). An overall condition $q$ is considered to be a sequence of elementary conditions $p_{i=1,n}$ accompanied by importance weights (called here *priorities*). It is interpreted in such a way that "$p_1$ should be satisfied (with priority 1) and among the solutions meeting $p_1$ (if any) the ones satisfying $p_2$ are preferred (with priority $\alpha_2$), and among those satisfying both $p_1$ and $p_2$, those satisfying $p_3$ are preferred with priority $\alpha_3$ ($\alpha_3 < \alpha_2 < 1$) and so on". This may be interpreted as nested implication operators: $p_1 \rightarrow (p_2 \rightarrow (p_3 \rightarrow \ldots$. The overall matching degree (the results of the aggregation) may be thus represented by the following membership function defining a fuzzy set of elements (rows) satisfying $q$ (when $q$ consists of 3 partial predicates):

$$\gamma(q,t) = \min\begin{pmatrix} \gamma(p_1,t), \max(\gamma(p_2,t), 1 - \min(\gamma(p_1,t),\alpha_2)), \\ \max(\gamma(p_3,t), 1 - \min(\gamma(p_1,t),\gamma(p_2,t),\alpha_3)) \end{pmatrix} \tag{23}$$

where $\min(\gamma(p_1,t),\alpha_2)$ and $\min(\gamma(p_1,t),\gamma(p_2,t),\alpha_3)$ are the priority levels (corresponding to $w_i$ in equation (20)) of partial fuzzy conditions (predicates) $p_2$ and $p_3$, respectively. Hence, concerning the condition $p_2$, its priority is $\alpha_2$ if $p_1$ is fully satisfied and is zero if $p_1$ is not at all satisfied, which reflects the fact that $p_2$ is conditioned by $p_1$. Thus, this is another example of the variable importance weight, but this time depending on the satisfaction of the "preceding" partial condition (predicate). Notice, that the hierarchy (nesting) of the conditions is here meant rather in the same sense as in Bosc's approach than in Lacroix and Lavency's.

## 3. Possibility theory for incomplete data representation

Classical relational data model offers only limited means to represent incomplete information. Namely, the NULL may be used instead of an unknown value. However, our knowledge about certain value, while incomplete, is often still not completely missing. This is especially true if our knowledge is of the linguistic type. For example, let us consider the employees characterized by the attributes NAME and AGE. Using classical relational database model we can easily represent information that, e.g., "John is 25". However, what if we know only that "John is young"? It is an imprecise proposition because it does not assign a particular value to John's age. The fuzzy logic has been conceived to deal with exactly this type of propositions. Thus, it is assumed that such proposition provides a *possibility distribution* of values for a variable under consideration (here: John's age), associating with each possible value a number in the interval [0,1]. In this case, we say (Zadeh 1978) that the proposition of the type $p$ = "$X$ is young" induces a possibility distribution $\pi$ (the notation $\pi_X$ is often used to indicate what variable is considered):

$$X \text{ is young} \rightarrow \pi = \text{YOUNG} \tag{24}$$

or, equivalently:

$$\forall u \in \text{U} \qquad \pi(u) = \mu_{\text{YOUNG}}(u) \tag{25}$$

that is, the possibility that a certain $u \in U$ is an actual value of $X$ is equal to the $u$'s membership degree to the fuzzy set YOUNG, which models the fuzzy term *young*. This framework makes it possible to represent also such propositions as:

$$X \in [20,25] \rightarrow \pi(u) = 1 \text{ for } u \in [20,25] \text{ and } \pi(u) = 0 \text{ for } u \notin [20,25] \quad (26)$$

or

$$X \text{ is } \textit{unknown} \rightarrow \pi(u) = 1 \quad \forall u \in U \quad (27)$$

Knowing a possibility distribution $\pi_X$ we may be also interested what is the possibility that $X$'s value belongs to a set $A \subseteq U$. This leads to the concept of the *possibility measure*, i.e., function $\Pi$ such that:

$$\Pi: 2^U \rightarrow [0,1] \quad (28)$$

From the postulated properties of the possibility measure it is assumed that (in fact, usually we start with the concept and properties of the possibility measure and only then the notion of the possibility distribution is introduced):

$$\Pi(A) = \sup_{u \in A} \pi(u) \quad (29)$$

The possibility measure alone does not tell us enough about the actual value of $X$. Thus, it is usually supplemented with the possibility measure of the complement of $A$. More precisely, the *necessity measure*, N, is defined, expressing the "impossibility" of the set $\overline{A}$, i.e.:

$$N(A) = 1 - \Pi(\overline{A}) = \inf_{u \in A} \pi(u) \quad (30)$$

This may be easily extended to the case where $A$ is a fuzzy set. Then:

$$\text{Possibility}(X \text{ is } A) = \Pi(A) = \sup_{u \in U} \min(\pi(u), \mu_A(u)) \quad (31)$$

and:

$$\text{Necessity}(X \text{ is } A) = N(A) = \inf_{u \in U} \max(1 - \pi(u), \mu_A(u)) \quad (32)$$

Now, if we know that the possibility distribution of the $X$'s value is $\pi$ then the degree to which the actual value of $X$ belongs to $A$ (often denoted as "$X$ is $A$") belongs to the interval $[N(A), \Pi(A)]$.

Prade and Testemale (1984) generalize the concept of the relational database in such a way that the value $A(t)$ of a tuple $t$ for an attribute $A$ may be given as a possibility distribution defined on the domain of this attribute (extended by adding a special element meaning "*inapplicable*"). They adapted classical relational algebra to the case of the possibilistic database. In order to illustrate the algebra, we discuss only the selection and give a relevant example of the query. The selection $\sigma(R, C)$ of a relation $R$ upon the condition $C$ may refer to two types of atomic conditions for $C$:

I. $A \theta a$, where $A$ is the name of an attribute, $\theta$ is a comparison operator (fuzzy or not) and $a$ is a constant (fuzzy or not); (33)

II. $A \theta B$, where $B$ is also an attribute name. (34)

More complex conditions may be built using the above atomic conditions and the logical connectives of negation, disjunction and conjunctions.

The matching degree of an atomic query and a tuple $t$ is computed as a pair: possibility and necessity measure (with respect to the possibility distributions $A(t)$ and $B(t)$) of relevant sets. In case of (I.) it is the set, crisp or fuzzy, of the elements belonging to the domain of $A$ and being in relation $\theta$ (crisp or fuzzy) with the constant $a$. In the second case (II.) it is the subset of the Cartesian product of domains of $A$ and $B$ containing only the pairs of elements being in relation $\theta$. In this case a joint possibility distribution over the Cartesian product of the domains of $A$ and B is used.

Formally, the matching degree for the case (I.) is computed as follows. Let us denote with $F$ the set (in general fuzzy) whose possibility and necessity measures have to be computed. Its membership function for the elements of the domain of the attribute $A$ is:

$$\mu_F(d) = \sup_{d' \in D} \min(\mu_\theta(d,d'), \mu_a(d')) \quad d \in \mathrm{Dom}(A) \tag{35}$$

Now, the possibility and necessity measures of the set $F$ with respect to the possibility distribution $\pi_{A(t)}$ being the value of the attribute $A$ for the tuple $t$ are computed as usually:

$$\Pi_{A(t)}(F) = \sup_{d \in D} \min(\pi_{A(t)}(d), \mu_F(d)) \tag{36}$$

$$\mathrm{N}_{A(t)}(F) = \inf_{d \in D} \max(1 - \pi_{A(t)}(d), \mu_F(d)) \tag{37}$$

In case (II.) the set $F$ comprises the pairs of elements $(d, d')$, $d \in \mathrm{Dom}(A)$, $d' \in \mathrm{Dom}(B)$ such that $d \; \theta \; d'$ is satisfied. Thus, its membership function is identical with that of $\theta$:

$$\mu_F(d,d') = \mu_\theta(d,d') \tag{38}$$

This time we have to compute the possibility and necessity measures with respect to a joint possibility distribution $\pi_{(A(x),B(x))}$ :

$$\pi_{(A(t),B(t))}(d,d') = \min(\pi_{A(t)}(d), \pi_{B(t)}(d')) \tag{39}$$

Then, the possibility and necessity measures are computed as previously:

$$\Pi_{(A(t),B(t))}(F) = \sup_{d \in D} \min(\pi_{A(t)}(d), \pi_{B(t)}(d'), \mu_F(d,d')) \tag{40}$$

$$\mathrm{N}_{(A(t),B(t))}(F) = \inf_{d \in D} \max(1 - \pi_{A(t)}(d), 1 - \pi_{B(t)}(d'), \mu_F(d,d')) \tag{41}$$

This type of the querying and the matching degree computation is referred to as *fuzzy pattern matching* (Dubois, Prade 1995). Using the previous notation: the pattern is a fuzzy set $F$ and the matching data sought is represented by a possibility distribution $A(t)$.

Bosc et al. (2000) introduced a new type of queries for possibilistic databases that do not rely on the fuzzy pattern matching. In this approach the parameters (syntax) of the possibility distributions are compared rather than their semantics. The formulas for the matching degree calculation refer to the following three auxiliary functions:

$$\mathrm{Poss}(A(t),\{d_1, \ldots, d_n\}) = \min(\pi_{A(t)}(d_1), \ldots, (\pi_{A(t)}(d_n)) \tag{42}$$

$$\mathrm{Card\_cut}(A(t),\lambda) = |\{d \in D: \pi_{A(t)}(d) \geq \lambda\}| \tag{43}$$

$$\mathrm{Card\_supp}(A(t)) = |\{d \in D: \pi_{A(t)}(d) > 0\}| \tag{44}$$

where $d_1, \ldots, d_n \in \mathrm{Dom}(A)$, and $\lambda \in [0,1]$. A value of the function $\mathrm{Poss}(A(t), \{d_1, \ldots, d_n\})$ may be interpreted as the truth degree of the statement "all the values $d_1, \ldots, d_n$ are possible for $A(t)$". The values of the functions $\mathrm{Card\_cut}(A(t), \lambda)$ and $\mathrm{Card\_supp}(A(t))$ correspond to the number of elements for which the possibility distribution $A(t)$ takes on a value above or equal to $\lambda$ and 0, respectively.

Then, we can compute a matching degree for the queries of the type (Bosc et al. 2000): "Find the houses for which the price value $100.000 is considered *more possible* than the value $80.000" or "Find the houses for which $100.000 is the *only* price value which is *completely possible*", which can be expressed, respectively, using:

$$\mathrm{Poss}(\mathrm{PRICE}(t),\{100.000\}) \geq \mathrm{Poss}(\mathrm{PRICE}(t),\{80.000\})$$

$$\mathrm{Poss}(\mathrm{PRICE}(t),\{100.000\}) = 1 \text{ and } \mathrm{Card\_cut}(\mathrm{PRICE}(t),1) = 1.$$

In order to use two possibility distributions in the same query we need some indices to compare them. Obviously, one candidate for such an index is the semantic comparison defined with (40) and (41), but some other alternative approaches do exist. Raju and Majumdar (1988) assume a resemblance relation defined on the interval [0,1]. Then, their *fuzzy equality measure* is defined as follows:

$$\mu_{EQ}(\pi_A, \pi_B) = \min_{u \in D} \psi(\pi_A(u), \pi_B(u)) \quad (45)$$

where $\pi_A$, $\pi_B$ are two possibility distributions on $D$ to be compared and $\psi$ is a resemblance relation (reflexive and symmetric) defined on [0,1]. Bosc et al. (2000) generalize this measure. They assume the existence of a resemblance relation *RES* defined on the domain *D*, upon which compared possibility distributions are defined. Their measure of the fuzzy equality between two possibility distributions is based both on the resemblance relation *RES*, and the resemblance relation over [0,1], named in their approach a proximity relation and denoted with $\theta$:

$$\mu_{s(A,B)}(u) = \sup_{v \in D} \min(\mu_{RES}(u,v), \mu_{\theta}(\pi_A(u), \pi_B(v)) \quad (46)$$

The equation above measures the degree to which the possibility distribution $\pi_A$ can replaced by the possibility distribution $\pi_B$ with respect to an element $u$ belonging to the support of $\pi_B$. Such an replacement is acceptable (the computed degree is high) if there exist $v$ belonging to the support of $\pi_B$ such that $u$ and $v$ are similar (in the sense of *RES*) and the values $\pi_A(u)$ and $\pi_B(v)$ are similar (in the sense of $\theta$). Then, the degree in which we can replace a possibility distribution $\pi_A$ with a possibility distribution $\pi_B$ with respect to the whole domain $D$ is given by the following equation:

$$\mu_{repl}(A,B) = \inf_{u \in D} \max(1-\pi_A(u), \mu_{s(A,B)}(u)) \quad (47)$$

Finally, the fuzzy equality measure between two possibility distributions $\pi_A$ and $\pi_B$ (equated here with A and B, respectively) is given by:

$$\mu_{EQ}(A,B) = \min(\mu_{repl}(A,B), \mu_{repl}(B,A)) \quad (48)$$

Hence, a condition of a representation-based (syntactic) query involving two possibility distributions can be expressed as:

$$\text{REP}(A) \approx \text{REP}(B) \quad (49)$$

where REP($A$) and REP($B$) are the representations (possibility distributions) of the values of attributes $A$ and $B$, respectively, which have the same domain $D$ or fuzzy sets defined on $D$ and used in a query (or more precisely possibility distributions induced by such fuzzy sets). For example, the condition *REP*(AGE) ≈ *REP*(*middle_aged*) will be true to a degree to which a value of attribute AGE (possibility distribution) is syntactically similar to the possibility distribution induced by a fuzzy set corresponding to the *middle_aged* concept. Notice, that in the case of fuzzy pattern matching a similar query may be used. However, it would produce a possibility/necessity measures of the event that the value of the AGE attribute belongs to fuzzy set *middle_aged* provided that all we know about the age is a possibility distribution.

## References

Bosc P., Pivert O. (1992) Fuzzy querying in conventional databases. In: *Fuzzy logic for the management of uncertainty*, L.A. Zadeh and J. Kacprzyk, eds. John Wiley & Sons.

Bosc P., Pivert O. (1993) An approach for a hierarchical aggregation of fuzzy predicates. In: *2nd IEEE Int. Conf. on Fuzzy Systems* (FUZZ-IEEE'93), USA.

Bosc P., Pivert O. (1995) SQLf: A Relational Database Language for Fuzzy Querying. *IEEE Trans. on Fuzzy Systems*, **3**, 1-17.

Bosc P., Duval L., Pivert O. (2000) Value-based and representation-based querying of possibilistic databases. In: *Recent issues on Fuzzy Databases*, G. Bordogna and G. Pasi, eds., Physica-Verlag, Heidelberg and New York, 3-27.

Codd E.F. (1970) A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM*, **13**(6), 377-387.

Date C.J. (1995) *An Introduction to Database Systems*. Addison-Wesley Publishing Company.

Dubois, D. and H. Prade (1995) Tolerant fuzzy pattern matching: an introduction. W: *Fuzziness in Database Management Systems*, P. Bosc and J. Kacprzyk, eds. Physica-Verlag, Heidelberg, 42-58.

Dubois D., Prade H. (1997) Using fuzzy sets in flexible querying: why and how?. In: *Flexible Querying Answering Systems*, T. Andreason, H. Christiansen, H.L. Larsen, eds. Kluwer Academic Publishers.

Kacprzyk J., Ziółkowski A. (1986) Database queries with fuzzy linguistic quantifiers. *IEEE Trans. on Syst., Man and Cybern. SMC*, 16, 474-479.

Kacprzyk J., Zadrożny S., Ziółkowski A. (1989) FQUERY III+: a "human-consistent" database querying system based on fuzzy logic with linguistic quantifiers. *Information Systems*. **14**, 443-453.

Kacprzyk J., Zadrożny S. (1995) FQUERY for Access: Fuzzy Querying for Windows-Based DBMS. In: *Fuzziness in Database Management Systems*, P. Bosc and J. Kacprzyk, eds. Physica-Verlag, Heidelberg. 415-433.

Kacprzyk J., Zadrożny S. (1997) Implementation of OWA operators in fuzzy querying for Microsoft Access. In: *The Ordered Weighted Averaging Operators: Theory and Applications*, R.R. Yager and J. Kacprzyk, eds. Kluwer, Boston. 293-306.

Lacroix M., Lavency P.(1987) Preferences: Putting more knowledge into queries. *Proc. of the 13 International Conference on Very Large Databases*. Brighton, UK, 217-225.

Petry F.E. (1996) *Fuzzy Databases: Principles and Applications*. Kluwer Academic Publishers, Boston, Dordrecht, London.

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.

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**, 129-166.

Tahani V. (1977) A Conceptual Framework for Fuzzy Query Processing: a step toward very Intelligent Database Systems. *Inf. Proc. and Manag.*, **13**, 289-303.

Yager R.R. (1988) On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man and Cybernetics*. **18**, 183-190.

Zadeh L.A. (1983) A Computational Approach to Fuzzy Quantifiers in Natural Languages. *Computational Mathematics Applications*. **9**, 149-184.

Zadeh L.A. (1978) Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*. **1**, 3-28.

ISBN 83-85847-78-2