

# Modeling Soft Conditions with Unequal Importance in Fuzzy Databases based on the Vector $p$ -norm

Gloria Bordogna

CNR IDPA  
via Pasubio 5, I-24044 Dalmine (BG) (Italy)  
gloria.bordogna@idpa.cnr.it

Giuseppe Psaila

University of Bergamo-Engineering Faculty  
via Marconi 5, I-24044 Dalmine (BG) (Italy)  
psaila@unibg.it

## Abstract

In this paper a modeling of soft selection conditions with preferences in fuzzy databases is proposed based on the vector  $p$ -norm operator. We outline the semantics of the compound query when the selection conditions are ANDed and ORed for increasing values of the parameter  $p$ .

**Keywords:** priorities, flexible querying, fuzzy databases.

## 1 Introduction

Flexible querying of databases is recognized as a useful feature to aid users in finding what they need. When a traditional SQL query is performed on a relational database, often two unpleasant scenarios may occur: either an empty answer is obtained as a result of a too selective query, or too many undifferentiated answers (records or tuples) are retrieved. A flexible query allowing the specification of preferences on attributes values of selection conditions (i.e., tolerant or soft selection conditions), and relative appraisal between conditions, intended as preference weights of conditions, is a useful means to produce discriminated answers.

Several approaches to model preferences in queries have been proposed in the literature [7] [9] [10] [14] [15] [18] [19]. [22] [23] [24] [26] [27] [30].

In fuzzy databases, to make possible the expression of preferences on the values of each attribute, the usual crisp selection conditions in queries have been relaxed to soft selection conditions, admitting satisfaction or confidence

degrees, i.e., interpreted as soft constraints that should be fulfilled as closely as possible [6] [15] [27]. Other approaches are based on either the explicit expression of order relations between the values of an attribute [10], or, the specification of ideal attribute values and a distance measure to rank the items in *top-k* queries [8].

The second aspect of a flexible query language is the possibility to express preferences between distinct conditions, intended as relative appraisal of conditions that tells to what extent a condition is preferred to another. In the following, we will name these preferences between conditions with the term *importance* or *priority* of the conditions. The modeling of conditions with distinct importance is strictly related to the problem of condition aggregation [1] [2] [3] [5] [6] [11] [13] [15] [25] [27] [31]. When expressing queries with conditions having unequal importance weights, either explicit importance weights are specified, or a cascading of the conditions is assumed so that the importance decreases from the first condition to the last one listed in the query. Furthermore, in the aggregation it is often implicitly assumed that the quantity of the important conditions that are met by an item in the database must positively increase the overall relevance of an item (*Pareto principle*) [1][22][23][24]. This behavior has been modeled by ranking the items basically by a weighted averaging of the satisfactions of the selection conditions, for instance, by the use of OWA operators with the semantics of fuzzy monotone increasing quantifiers, thus achieving a compromise between several criteria [1] [2] [21] [20] [25] [32].

In this paper we analyze the distinct semantics of soft selection conditions with unequal importance in fuzzy databases when they are combined by the AND and OR aggregator, and we propose a definition of ANDED and ORED query conditions based on the use of the  $p$ -norm operator [29]. We show that this approach is flexible: it allows modeling distinct behavioral properties of the aggregation (namely distinct semantics for AND and OR). ANDED (resp. ORED) soft conditions with unequal importance weights behave as *veto conditions*, necessary to satisfy (resp., as *favor conditions*, sufficient to have some satisfied) [17].

The paper is organized as follows: in the next section the basic idea of our approach is illustrated. In section 3 the evaluation of soft conditions with importance weights is formalized. In section 4 an example of flexible query with soft conditions is discussed. Finally, the conclusions summarize the main results.

## 2 Basic Ideas of the Proposal

When specifying multiple conditions with unequal importance in a database query, distinct interpretations of the aggregation can be assumed. One may consider an exclusive meaning like in the request “*find early departure flights (most important) or early arrival trains (less important) to Rome*”. This query expresses the interest for either flights or trains to Rome so that flights must be ranked first. On the contrary, the query “*find a house to rent that is cheap (most important), big (important), and recent (fairly important)*” expresses the interest for a house meeting all the conditions in the order specified by the importance weights. In this case the implicit aggregator is an AND. Between these two extreme cases there may be requests in which one wants to model mixed aggregations and softer interpretations of either AND and OR, such as in “*find early departure, cheap, safe, flights or early arrival, comfortable fast trains to Rome*” which demand for a trade off between the degrees of satisfaction of the most important conditions and the number of the conditions satisfied.

The proposal presented in this paper takes inspiration from a Boolean extended retrieval model [29]. In this IR model a query is a list of weighted terms, in which the weight specifies the importance of the search term, and the vector space model is assumed as the basic retrieval model for computing the relevance scores of the documents. The peculiarity of the model is that the query can be a Boolean query and the similarity measures used to compute the relevance of a document to queries with ANDED or ORED terms are defined in terms of a *vector p-norm*, so that for distinct values of the parameter  $p$  different semantics of the aggregations AND and OR are modelled.

We adapt this idea to the modeling of soft query conditions with distinct importance in classic relational databases. To this end, we replace the search terms of the IR query by soft selection conditions expressed by linguistic predicates that specify preferences over the attribute values as in fuzzy queries. The linguistic predicate semantics are defined by fuzzy subsets of the attribute domains and are interpreted as fuzzy constraints on the values of the attributes [6] [12] [15] [19] [27]. This way, the comparability of satisfaction degrees of soft conditions, i.e., of confidence degrees defined in  $[0,1]$ , is achieved for any kind of attribute domain (both numeric cardinal or ordinal and nominal).

The adoption of a  $p$ -norm operator in the context of databases is not completely new. In [8], an evaluation mechanism of *top-k* queries to relational databases, aggregated by a  $p$ -norm operator has been proposed. In this approach, preferences over the attribute values are specified by ideal attribute values. The ranking of a tuple with respect to each condition is based on a distance measure (as for usual *top-k* queries). However, in this approach unequal importances of the conditions are not modeled, while this is the objective of our proposal.

## 3 Semantics of ANDED and ORED Soft Conditions with Importance Weights

Let us consider the following two flexible queries with two soft selection conditions,  $A1$  and  $A2$ , over a classic relational database:

$$q_{AND}=A1 \text{ AND } A2 \quad q_{OR}=A1 \text{ OR } A2$$

We assume to model the soft selection conditions  $A1$  and  $A2$  as fuzzy constraints on the domain of some attribute:

for example,  $A1=Big$  defined as a fuzzy constraint on the dimension domain of flats (see Figure 1) and  $A2 =Cheap$  on the domain of the price of a flat (see Figure 2). The evaluation of each soft condition  $A$  by an attribute value  $v$  of a tuple  $t$  (a flat) is a value  $\mu_A(t) \in [0,1]$ .

We can represent the tuples as points in a two dimensional space, in which each axis corresponds to a soft condition, and the coordinates of a point  $t$  are the respective degrees of satisfaction of the soft conditions by tuple  $t$ . In other words, tuple  $t$  is represented by point  $(\mu_{A1}(t), (\mu_{A2}(t))$ .

Aggregate queries can be represented in this space as well:  $q_{AND} = A1 \text{ AND } A2$  identifies an ideal tuple corresponding with point (1,1) (see Figure 3). In fact, this request is best satisfied by tuples that fully satisfy both conditions. Furthermore, it is partially satisfied by tuples that correspond to points close to (1,1) and their satisfaction degree  $sim(t, A1 \text{ AND } A2)$  decreases with their distance from (1,1):

$$sim(t, A1 \text{ AND } A2) \equiv 1 - \sqrt{\frac{(1 - \mu_{A1}(t))^2 + (1 - \mu_{A2}(t))^2}{2}} \quad (1)$$

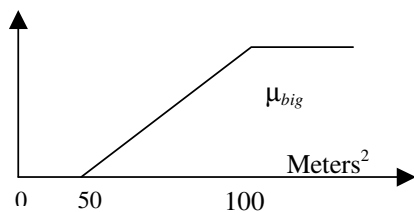
The two tuples  $t1$  and  $t2$  represented in Figure 3 get a ranking:

$$sim(t2, A1 \text{ AND } A2) > sim(t1, A1 \text{ AND } A2)$$

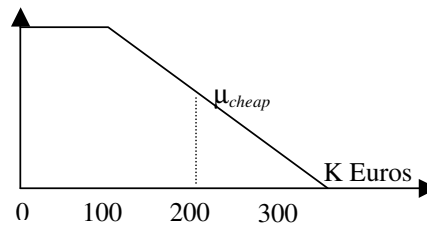
On the other side,  $q_{OR} = A1 \text{ OR } A2$  identifies an undesired tuple corresponding with point (0,0) (see Figure 4): in fact, this request is not satisfied only by tuples that do not satisfy any of the two conditions, while all the tuples satisfying, at least a little, a soft condition are acceptable; the satisfaction degrees  $sim(t, A1 \text{ OR } A2)$  increase with their distance from (0,0):

$$sim(t, A1 \text{ OR } A2) \equiv \sqrt{\frac{\mu_{A1}(t)^2 + \mu_{A2}(t)^2}{2}} \quad (2)$$

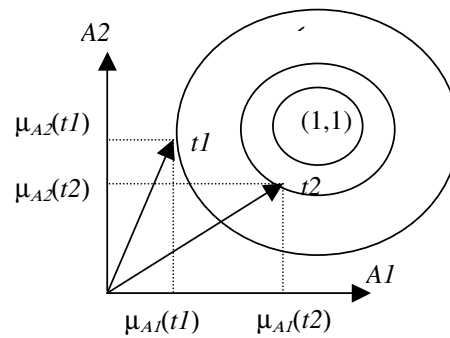
For tuples  $t1$  and  $t2$  represented in Figure 4, it is  $sim(t2, A1 \text{ OR } A2) > sim(t1, A1 \text{ OR } A2)$ .



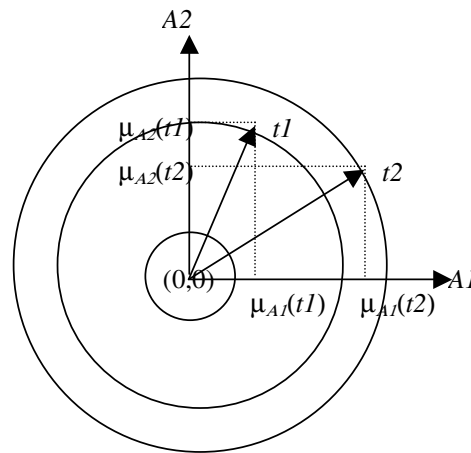
**Figure 1:** semantics of *big* defined as a fuzzy set on the dimension in square meters of flats



**Figure 2:** semantics of *cheap* defined as a fuzzy set on the price in K Euros of flats



**Figure 3:** Equidistant lines from (1,1) point representing the “ANDed” selection conditions  $q_{AND} = A1 = big \text{ flats AND } A2 = cheap \text{ flats}$



**Figure 4:** Equidistant lines from (0,0) point for “ORed” selection conditions  $q_{OR} = A1 = big \text{ flats OR } A2 = cheap \text{ flats}$

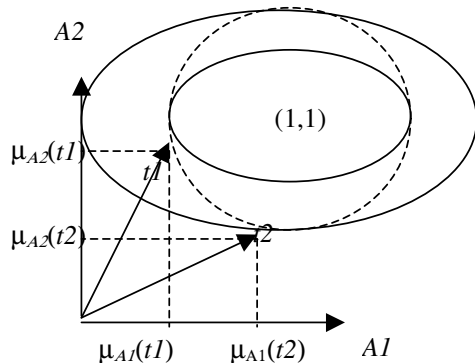
In the case in which the soft selection conditions have unequal priorities expressed by importance weights  $i_1$  and  $i_2$  respectively, with  $i_1, i_2 \in [0,1]$ , the formulas (1) and (2) can be modified as follows:

$$sim(t, A1, i_1 \text{ AND } A2, i_2) \equiv 1 - \sqrt{\frac{i_1^2 (1 - \mu_{A1}(t))^2 + i_2^2 (1 - \mu_{A2}(t))^2}{i_1^2 + i_2^2}} \quad (3)$$

$$\text{sim}(t, A1, i_1 \text{ OR } A2, i_2) \equiv 1 - \sqrt{\frac{i_1^2 (\mu_{A1}(t))^2 + i_2^2 (\mu_{A2}(t))^2}{i_1^2 + i_2^2}} \quad (4)$$

When the conditions are fully important, it is  $i_1=i_2=1$ , the similarity measures in formulae (3) and (4) reduce to formulae (1) and (2), i.e., the equidistant lines in the bidimensional space are still circles centered in (1,1) for ANDED conditions and in (0,0) for ORED conditions. When one condition is not important at all, its importance weight is zero; then the similarity measures in formulae (3) and (4) is determined solely by the other condition.

If  $i_2 > i_1$ , it means that a tuple satisfying condition A2 is preferred to a tuple satisfying A1, and in this situation the equidistant lines in the bi-dimensional space are no more circles but ellipses centered in (1,1) for ANDED conditions (the dotted circle becomes the continuous ellipses in Figure 5) and centered in (0,0) for ORED conditions. In Figure 5, it can be noticed that since A1 has a lower priority than A2, then tuple  $t1$  gets now a higher ranking with respect to tuple  $t2$  (the complement of the normalized distance from (1,1)).



**Figure 5:** Equidistant lines from (1,1) for  $q_{\text{AND}} = A1 = \text{big flats}, i_1 \text{ AND } A2 = \text{cheap flats}, i_2$  with  $i_1 < i_2$

#### 4 Formalization of Prioritized Soft Conditions with a $p$ -norm Operator

In all the formulae above, we have assumed a normalized Euclidean distance at the basis of the computation of the similarity measure. This distance can be generalized to a  $n$ -dimensional space based on the *Minkowski* distance. Given a vector  $V = [v_1, \dots, v_n]$  with  $v_k \in [0,1]$  and  $k=1, \dots, n$ , a  $p$ -norm operator is defined for as follow:

$$\|V\|_p \text{ normalized} = \left(\frac{1}{n}\right)^{\frac{1}{p}} \|V\|_p = \left[ \frac{\sum_{k=1}^n v_k^p}{n} \right]^{\frac{1}{p}} \quad (5)$$

with  $1 \leq p \leq \infty$ .

When  $p=2$  formula (5) in a bi-dimensional space is the Euclidean distance.

When associating unequal importance weights  $i_1, \dots, i_n \in [0,1]$  with the conditions, the effect of satisfying (or not satisfying) a condition should decrease as the importance of satisfying the condition decreases.

In the general case of  $n$  ANDED soft conditions with unequal priorities  $i_1, \dots, i_n \in [0,1]$ :

$$q_{\text{AND}}^p = A1, i_1 \text{ AND}^p A2, i_2 \text{ AND}^p \dots An, i_n$$

the ranking score of a tuple  $t$  can be determined based on a generalization of (3) as follows:

$$\text{sim}(t, q_{\text{AND}}^p) \equiv 1 - \left[ \frac{\sum_{k=1}^n i_k^p (1 - \mu_{A_k}(t))^p}{\sum_{k=1}^n i_k^p} \right]^{\frac{1}{p}} \quad (6)$$

In the general case of  $n$  ORED soft conditions with unequal priorities  $i_1, \dots, i_n$ :

$$q_{\text{OR}}^p = A1, i_1 \text{ OR}^p A2, i_2 \text{ OR}^p \dots An, i_n$$

the ranking score of a tuple is determined as follows:

$$\text{sim}(t, q_{\text{OR}}^p) \equiv \left[ \frac{\sum_{k=1}^n i_k^p (\mu_{A_k}(t))^p}{\sum_{k=1}^n i_k^p} \right]^{\frac{1}{p}} \quad (7)$$

Some considerations on the behavioral properties of the similarity measures can be made depending on the different settings of the parameter  $p$ .

When  $p=1$ , we have the *Hamming* distance, and it can be easily proved that formulae (6) and (7) reduce to the same one:

$$\text{sim}(t, q_{\text{OR}}^1) \equiv \text{sim}(t, q_{\text{AND}}^1) \equiv \left[ \frac{\sum_{k=1}^n i_k (\mu_{A_k}(t))^1}{\sum_{k=1}^n i_k} \right]^{\frac{1}{1}}$$

that is, the meaning of AND and OR is no more modeled but the two queries are evaluated by

the same evaluation function corresponding to the vector space model of Information Retrieval [28]. In this case, the equidistant lines from (1,1) and (0,0) points in a bidimensional space are parallel lines perpendicular to the main diagonal of the first quadrant. This aggregation is a weighted arithmetic mean that has been proved in [25] to be order equivalent to an aggregation based on the OWA operator with weighting vector  $w_i=1/n \quad \forall i=1,..,n$  and the same importance weights  $i_1, \dots, i_n$  of the operands. This OWA operator has an *Orness* equal to 0.5, which corresponds to an aggregation behavior exactly intermediate between AND and OR.

When  $p=\infty$  and all the conditions have equal priority, we obtain:

$$\begin{aligned} \text{sim}(t, q_{\text{AND}}^\infty) &= \min(\mu_{A_1}(t), \dots, \mu_{A_n}(t)) \\ \text{sim}(t, q_{\text{OR}}^\infty) &= \max(\mu_{A_1}(t), \dots, \mu_{A_n}(t)) \end{aligned}$$

This is the situation modeled within fuzzy sets, in which the AND and the OR are associated with *min* and *max*, respectively. With this setting of  $p$ , we can model opposed tendencies of the aggregation, conjunction and disjunction oriented. In the case of AND, the ranking of tuples takes into account all the conditions, i.e., we consider the soft conditions as veto conditions, necessary to satisfy. In the case of OR, the ranking of tuples takes into account at least a condition, thus we consider the soft conditions as favor conditions that is sufficient to have one satisfied [17].

When  $p=\infty$  and the conditions have unequal importance, we have:

$$\begin{aligned} \text{sim}(t, q_{\text{AND}}^\infty) &\approx 1 - \frac{\max_{i=1,..,n} [i_1(1-\mu_{A_1}(t)), \dots, i_n(1-\mu_{A_n}(t))]}{\max_{i=1,..,n} (i_k)} = \\ &= \min \left[ 1 - \frac{i_1}{\max_{i=1,..,n} (i_k)} (1-\mu_{A_1}(t)), \dots, 1 - \frac{i_n}{\max_{i=1,..,n} (i_k)} (1-\mu_{A_n}(t)) \right] \end{aligned}$$

This corresponds to applying an importance weighted transformation function defined for AND aggregation  $H_{\text{AND}}(\omega, \mu) = \neg \omega \oplus \mu$  in which  $\oplus$  is the *t-conorm* algebraic sum,  $\omega = i / \max_{i=1,..,n} (i_k)$  is the normalized importance weight of  $\mu$ , and  $\mu$  the degree of satisfaction of a condition:  $H_{\text{AND}}(\omega, \mu) = \neg(\omega(-\mu)) = 1 - \omega(1-\mu)$  [25].

Analogously, for the ORed prioritized conditions and  $p=\infty$  we obtain the following:

$$\text{sim}(t, q_{\text{OR}}^\infty) = \frac{\max_{i=1,..,n} [i_1(\mu_{A_1}(t)), \dots, i_n(\mu_{A_n}(t))]}{\max_{i=1,..,n} (i_k)}$$

This corresponds to applying the importance weighted transformation function defined for the OR aggregations  $H_{\text{OR}}(\omega, \mu) = \omega \otimes \mu$ , in which  $\otimes$  is the *t-norm* algebraic product [25].

These results allow us to state that the aggregation of prioritized soft query conditions based on a *p-norm* subsumes the fuzzy models in which the aggregations are based on OWA operators with weighting vectors  $W$  ( $w_i=1/n, \forall i=1,..,n$ ),  $w_1=1$  (*max*, i.e., OR) and  $w_n=1$  (*min*, i.e., AND).

By increasing  $p$  above 1, for example 2, 3 etc., one can model different types of aggregation behaviors that are partially compensative, in which we distinguish more and more the semantics of AND and OR, thus reinforcing the veto and favor semantics of soft conditions, respectively. The larger the value of  $p$  the more emphasis is given to the query structure specified by the aggregation operators AND and OR. On the contrary, by decreasing  $p$ , the distinction between the aggregation behaviors of the AND and OR becomes looser and disappears when  $p=1$ .

By assuming all importance weights equal to 1, the ranking scores that can be obtained by varying  $p$  between 1 and  $\infty$  are intermediate between the extreme cases when  $p=1$  and  $p=\infty$ :

$$\begin{aligned} \text{sim}(t, q_{\text{AND}}^\infty) &\leq \text{sim}(t, q_{\text{AND}}^p) \leq \text{sim}(t, q_{\text{AND}}^1) = \\ &= \text{sim}(t, q_{\text{OR}}^1) \leq \text{sim}(t, q_{\text{OR}}^p) \leq \text{sim}(t, q_{\text{OR}}^\infty) \end{aligned}$$

The strict equality holds when all soft conditions have the same degree of satisfaction and same importance weight 1. The intuition suggests that for an intermediate value of  $1 < p < \infty$ , the similarity functions in formulae (6) and (7), defined for conjunctive  $q_{\text{AND}}^p$  and disjunctive queries  $q_{\text{OR}}^p$ , correspond to distinct averaging operators modeling veto-towards and favor-towards behaviors of the aggregation, respectively [17].

Further, it can be proved that the following properties hold:

*Commutativity:*

$$A1, i_1 \text{ AND}^p A2, i_2 = A2, i_2 \text{ AND}^p A1, i_1$$

$$A1, i_1 \text{ OR}^p A2, i_2 = A2, i_2 \text{ OR}^p A1, i_1$$

*Duality:*

$$\text{NOT} [A1, i_1 \text{ OR}^p A2, i_2] = \text{NOT}[A1, i_1] \text{ AND}^p \text{NOT}[A2, i_2]$$

$$\text{NOT} [A1, i_1 \text{ AND}^p A2, i_2] = \text{NOT}[A1, i_1] \text{ OR}^p \text{NOT}[A2, i_2]$$

*Involution:*

$\text{NOT}[\text{NOT} [A1,i_1 \text{ AND}^p A2,i_2]] = [A1,i_1] \text{ AND}^p [A2,i_2]$

$\text{NOT}[\text{NOT} [A1,i_1 \text{ OR}^p A2,i_2]] = [A1,i_1] \text{ OR}^p [A2,i_2]$

*Idenpotency:*

$A,i_1 \text{ AND}^p A, i_2 = A; A,i_1 \text{ OR}^p A,i_2 = A$

While *associativity* is not completely satisfied:

$[[A1,i_1 \text{ AND}^p A2,i_2], i] \text{ AND}^p A3,i_3 =$

$[A1,i_1 \text{ AND}^p A2,i_2i] \text{ AND}^p A3,i_3(i_1+i_2)^{1/p}$

$[[A1,i_1 \text{ OR}^p A2,i_2], i] \text{ OR}^p A3,i_3 =$

$[A1,i_1 \text{ OR}^p A2,i_2i] \text{ OR}^p A3,i_3(i_1+i_2)^{1/p}$

and *distributivity* of Boolean expressions is not valid.

## 5 Example of Prioritized Soft Conditions Evaluation

Let us consider the relation shown in table 1, 1-4 columns, in which tuples represent distinct flats and their attributes are price, dimension and floor. Let us assume that the prioritized soft selection conditions are expressed in the *WHERE* clause of the basic Soft SQL query, an extension of SQL for allowing the expression of flexible queries to classic relational databases [4]:

```
SELECT TOP 5 C.Id,
FROM FLAT as C,
WHERE C.Price IS_priority cheap AND
      Dimension IS_priority big AND
      Floor IS_priority high
DEGREE THRESHOLD: 0.6;
```

The predicate *IS\_priority* expresses that the soft conditions *cheap* for price, *big* for dimension and *high* for floor have associated decreasing importance weights from the first one to the last one, so that the soft SQL *WHERE* clause can be rewritten as:

$q_{\text{AND}}^p = \text{Cheap},1 \text{ AND } \text{big},.5 \text{ AND } \text{high},.25$

For comparing the results, we consider also :

$q_{\text{OR}}^p = \text{Cheap},1 \text{ OR } \text{big},.5 \text{ OR } \text{high},.25$

In Table 1, columns 5-7, the degrees of satisfaction of soft conditions are shown. In Table 2, the results of  $q_{\text{AND}}^p$  and  $q_{\text{OR}}^p$  by distinct  $p$  values are reported. It can be seen that for  $p=1$  the two queries produce the same ranking scores. By increasing  $p$ , the ranking scores of columns t1-t3 decrease for AND and increase for OR, since the veto and favor behaviors of AND and OR aggregation semantics is reinforced and dominates the results. As  $p$  increases, this trends of results for AND and OR

aggregations is always true when all the conditions have equal importance. However, in the case of distinct importance weights, the ranking scores can be dominated by the effect of partial satisfaction of the most important conditions. This is what happens in the cases of tuples *t4* and *t5*, where the low satisfaction degree of the fully important condition *cheap* determines more heavily the ranking score of the full satisfaction degrees of the less important conditions. Further, for each given  $p>1$ , each tuple gets a greater ranking score when aggregated in OR than when aggregated in AND in accordance with the favor-towards and veto-towards behavior of the two types of queries. Finally, Table 3 shows the results of the evaluation of:

$q^p = \text{Cheap},1 \text{ AND } (\text{big},0,5 \text{ OR } \text{high},0,25)$

in which *cheap* price and the compound condition (*big* dimension OR *high* floor ) are considered as veto conditions, while the single conditions *big* dimension and *high* floor are considered as favor conditions. We assume that the implicit importance weight of the compound condition in parenthesis is the maximum of its single conditions, i.e., 0,5. However, this is a simple choice suggested by the fact that the two single conditions are Ored. A better choice would ask for a weight computed by taking into account the behavior semantics of the aggregation operator that is used. It can be noticed that, for tuple *t1* and *t2*, the ranking scores do not vary monotonically by increasing  $p$ , since, in this query, the ranking score is determined by a trade-off between the favor and veto semantics of the nested OR and AND which have opposed influence. Tuples *t3* and *t4* get increasing ranking scores as  $p$  increases, while *t5* gets decreasing values.

## 4 Conclusions

A model of prioritized soft conditions in flexible database queries based on the *p-norm* operator is proposed. While preferences on attribute values are defined by fuzzy constraints on the attribute domains, as in fuzzy databases, priorities are managed in the aggregation of soft selection conditions based on the *p-norm* operator. The advantage, with respect to previous approaches, is that the Boolean structure of the *WHERE* SQL clause can be maintained with the usual AND and OR connectives, but their semantics can be tuned by the parameter  $p>1$  to be more or less strict and distinct, thus subsuming as particular

cases both the Fuzzy aggregations  $AND=min$  and  $OR=max$  and the aggregation of the vector space model (inner product). Furthermore, for  $p>1$ , intermediate fuzzy aggregations can be modeled in a unifying framework with a veto-towards and favor-towards semantics when the conditions have equal importance. Moreover, one could mix in the same query AND and OR with distinct interpretations, i.e., characterized by distinct values of  $p$ . Finally by introducing importance weights, the aggregation behavior can be made more or less dependent on the satisfaction of the most important conditions.

**Table 1:** Classic relation representing Flats

flat	Price M€	Dim. m <sup>2</sup>	Floor #	$\mu_{cheap}$ 1	$\mu_{big}$ 0.5	$\mu_{high}$ 0.25
<i>t1</i>	90	100	4	1	1	0.5
<i>t2</i>	90	60	9	1	0.5	1
<i>t3</i>	100	80	8	1	0,5	0,5
<i>t4</i>	200	110	7	0,5	1	1
<i>t5</i>	300	110	9	0,3	1	1

**Table 2:** Ranking scores of  $q_{AND}^p$  and  $q_{OR}^p$  by distinct  $p$  values

	$q_{AND}^{p=1}$	$q_{AND}^{p=2}$	$q_{OR}^{p=2}$	$q_{AND}^{p=10}$	$q_{OR}^{p=10}$	$q_{AND}^{p=\infty}$	$q_{OR}^{p=\infty}$
<i>t1</i>	0,928	0,89	0,98	0,875	0,999	0,875	1
<i>t2</i>	0,857	0,781	0,925	0,750	0,999	0,75	1
<i>t3</i>	0,785	0,756	0,906	0,750	0,999	0,75	1
<i>t4</i>	0,714	0,563	<b>0,654</b>	0,500	<b>0,503</b>	0,5	<b>0,5</b>
<i>t5</i>	0,600	0,388	<b>0,553</b>	0,3	<b>0,5</b>	0,3	<b>0,5</b>

**Table 3:** Ranking scores of  $q_p = Cheap, 1 AND (big, .5 OR high, .25)$  by distinct values of  $p$

	$q^{p=1}$	$q^{p=2}$	$q^{p=5}$	$q^{p=10}$	$q^{p=50}$	$q^{p=100}$
<i>t1</i>	0,94	<b>0,922</b>	0,93	0,96	0,992	0,996
<i>t2</i>	0,89	<b>0,83</b>	0,85	0,90	0,978	0,989
<i>t3</i>	0,83	0,71	0,60	0,55	0,51	0,505
<i>t4</i>	0,66	0,59	0,54	0,51	0,504	0,502
<i>t5</i>	0,15	0,212	0,279	0,279	0,295	0,297

## References

[1] Agrawal R., Wimmers E.L. A framework for expressing and combining preferences,

proc. of the ACM SIGMOD 297-306, Dallas, 2000.

[2] Agrawal S., Chaudhuri S., Das G., Gionis A., Automated Ranking of Database Query Results, in Proc. Of 2003 CIDR Conference, 2003.

[3] Bordogna G., Pasi G., "A flexible approach to evaluating soft conditions with unequal preferences in fuzzy databases", Int. Journal of Intelligent Systems, 22, 665-689, 2007.

[4] Bordogna G., Psaila G., Customizable Flexible Querying for Classical Relational Databases, In J. Galindo (ed.) Handbook of Research on Fuzzy Information Processing in Databases. Hershey, PA, USA: Information Science Reference, 2008.

[5] Borzsonyi S., Kossmann D., Stocker K., The Skyline Operator, 17th International Conference on Data Engineering (ICDE'01), 421-430, 2001

[6] Bosc P, Liétard L., Pivert O., Rocacher D., Base de donnée - Gradualité et imprécision dans les bases de données, Technosup series, Ellipses, 2004.

[7] Bosc P., Pivert O., SQLf: a relational database language for fuzzy querying, IEEE Trans. On Fuzzy Systems, 3(1), 1995.

[8] Bruno, N., Chaudhuri, S., and Gravano, L. 2002. *Top-k* Selection Queries over Relational Databases: Mapping Strategies and Performance Evaluation. ACM Transactions on Database Systems 27, 2 (June), 153{187}, 2002.

[9] Buckles B., Petry F., A fuzzy model for relational Databases, Int. Journal of Fuzzy Sets and Systems, 7, 213-226, 1982.

[10] Chomicki J., Preference formulas in relational queries, ACM Trans. On Database Systems, 28(4), 427-466, 2003.

[11] De Tré G., De Baets B., Aggregating Constraint Satisfaction Degrees Expressed by Possibilistic Truth Values, IEEE Transactions on Fuzzy Systems, 11(3), 354-360, 2003.

[12] Dubois D. & Prade H. (1997) Using fuzzy sets in flexible querying: why and how?. In: Andreasen T, Christiansen H, Larsen HL (eds) Flexible query answering systems. Kluwer Academic Publishers, pp 45-60

- [13] Dubois D., Prade H., Testemale C., Weighted Fuzzy Pattern Matching ,Fuzzy Sets and Systems, 28, 313-331, 1988.
- [14] Eduardo J.; Goncalves M.; Tineo L. A Fuzzy Querying System based on SQLf2 and SQLf3, in Proc. of the XXX Conferencia Latino-Americana de Informática (CLEI 2004), Arequipa, Peru, September 27-October 1, 2004.
- [15] Galindo J., Medina J.M. & Aranda G.M.C. Querying fuzzy relational databases through fuzzy domain calculus. International Journal of Intelligent Systems, 14:375-411, 1999.
- [16] Galindo, J., Urrutia A., Piattini M, Fuzzy Databases, Modeling, Design and Implementation, Idea Group, 2006.
- [17] Grabisch M.,S. A. Orlovski, and R. R. Yager. Fuzzy aggregation of numerical preferences. In R. Slowinski, editor, *Fuzzy Sets in Decision Analysis, Operations Research and Statistics*, Kluwer Academic Publishers, Boston, MA, 31-68, 1998.
- [18] Ichikawa T., Hirakawa M., ARES: a relational database with the capability of performing flexible interpretation of queries, IEEE Trans. On Software Engineering, 12(5), 624-634, 1986.
- [19] Kacprzyk J. & Zadrosny S. (1995). FQUERY for access: fuzzy querying for windows-based DBMS. In P. Bosc and J. Kacprzyk, (eds), *Fuzziness in database management systems*. Physica verlag.
- [20] Kacprzyk J. & Zadrosny S. (1997) Implementation of OWA operators in fuzzy querying for Microsoft Access. In: Yager RR, Kacprzyk J (eds) *The ordered weighted averaging operators: theory and applications*. Kluwer, Boston, pp 293-306
- [21] Kacprzyk J. & Ziolkowski A. (1986). Database queries with fuzzy linguistic quantifiers. IEEE Transactions on Systems, Man and Cybernetics, 16: pp 474-479.
- [22] Kießling W., Foundation of Preferences in Databases Systems, in Proc. of the 28th VLDB Conf., Hong kong, China, 2002.
- [23] Kießling W., Kostler G., Preference SQL– Design, Implementation, Experiences. Proc. of the 28th VLDB Conf. Hong Kong, 2002.
- [24] Lacroix M., Lavency P., Preferences : putting more knowledge into queries, Proc. Of 13th Conf. VLDB, Brighton, GB, 217-225, 1987.
- [25] Larsen, H.L., Importance weighted OWA aggregation of multicriteria queries Larsen, H.L. , in Proc. Of NAFIPS 1999, New York, 740-744, 1999.
- [26] Motro A., "VAGUE: A User Interface to Relational Databases that Permits Vague Queries," ACM Trans. on Information Systems (TOIS), 6(3), pages 187--214, July 1988.
- [27] Petry F.E., *Fuzzy Databases*, (Kluwer Academic Pub., 1996).
- [28] Salton, G. and McGill, M.J. (1983). *Introduction to modern information retrieval*. New York, NY: McGraw-Hill.
- [29] Salton, G., Fox, E., and Wu, H. (1983). Extended Boolean information retrieval. *Communications of the ACM*, **26(12)**, 1022-1036.
- [30] Tahani V., A conceptual framework fro query processing: a step toward very intelligent database systems, Information Processing and Management, 13, 289-303, 1977.
- [31] Takaci, A., Skrbic, S., Srdjan Comparing priority, weighted and queries with threshold in PFSQL, Intelligent Systems and Informatics, 2007. SISY 2007. 24-25, Aug., 77 – 80, 2007.
- [32] Tineo L. (2000) Extending RDBMS for allowing Fuzzy Quantified Queries, in DEXA Proc. M. Kung (eds), LNCS 1873, Springer Verlag, 407-416.