Bipolar Queries: An Approach and its Various Interpretations

Sławomir Zadrożny[1] Janusz Kacprzyk[2]

1.Systems Research Institute, Polish Academy of Sciences
Warszawa, Poland
2.Systems Research Institute, Polish Academy of Sciences
Warszawa, Poland
Email: Slawomir.Zadrozny@ibspan.waw.pl, Janusz.Kacprzyk@ibspan.waw.pl

Abstract— *The concept of a bipolar query is studied in the framework of the flexible fuzzy querying of databases. The focus is on the aggregation of the negative and positive conditions forming a bipolar query. Three formal logical representations of such an aggregation are proposed and analyzed taking into account various possible interpretations of fuzzy logical connectives. The relation to other approaches known in the literature is shown.*

Keywords— aggregation operators, bipolar queries, flexible queries, fuzzy logic, logical connectives modeling.

1 Introduction

A query to a (relational) database may be identified, in a slightly simplified view, with a condition expressing what the user is looking for. A database management system returns in the response to a query a list of tuples satisfying this condition. Such a condition is usually composed of a few, simple, atomic conditions putting some constraints on the values of the attributes characterizing given relation (table). We will adopt such a simplified view, assuming moreover that the user is concerned with just one relation, i.e., using SQL's terminology, joins, subqueries etc. are excluded for the sake of clarity of the presentation of the main idea.

Atomic conditions are connected using the classical logical connectives of the conjunction, disjunction and negation. In many standard application scenarios this is exactly what is needed to retrieve required data from a database. However in some other scenarios more sophisticated forms of the queries seem to be useful. Thus, in the literature further extensions of this basic setting are proposed. Firstly, many authors advocate the convenience of using *linguistic terms* in queries (cf., e.g., [1, 2]). For example, it is more natural and comfortable for a user of a real-estate agency to state that she or he is looking for a "not very expensive house" rather than to use a precise interval of acceptable prices. The benefits of using fuzzy logic to model linguistic terms in queries are widely advocated and there is a large number of of both theoretical and practical results obtained by the researchers in this area (cf., e.g., [3, 1, 2]). Secondly, some (atomic) conditions may be for the user more important than the others. For example, a customer of a real-estate agency may be looking for a "not very expensive house located in a nice city district" and to treat the former condition as a much more important than the latter. Thus in the overall matching degree of a tuple against such a query the satisfaction of the former condition is crucial, while of the latter is to some extent secondary. Proper modeling of importance weights is a subject of many papers (cf., e.g., [4]).

The essence of the concept of a *bipolar query* is in a new way of differentiating the conditions than by assigning them with some *fixed* importance weights. Namely, a bipolar query is defined by two conditions: one – *negative (required)* – expresses the constraints that *have to be* satisfied while the second – *positive (preferred)* – expresses only what is *desired* and its violation does not necessarily lead to the rejection of given tuple. This combination of conditions may be meant and studied in a few different ways. Here we are interested in the ways the satisfaction of both types of conditions should be combined to obtain an overall matching degree of the whole query. From this point of view the most important is the question if there is a conflict between the conditions or they can be satisfied simultaneously. If there is a total conflict, i.e., satisfying one condition means totally failing to satisfy another, the bipolar query reduces itself to the required condition. On the other hand, if both conditions may be totally satisfied simultaneously then the bipolar query reduces to a simple conjunction. Thus, the most interesting are intermediate cases which may be characterized by a degree of conflict between the conditions.

In this paper we study three logical formulas which may be used as a representation of a bipolar query. They are equivalent in case of the classical predicate logic but as soon as we adopt the fuzzy (multivalued) logic context they become distinct and exhibit different properties depending on the assumed set of fuzzy operators used to model logical connectives. Our study may be seen as an attempt at justifying particular choices in this respect.

The structure of this paper is as follows. In Section 2 we briefly remind the concept of the bipolar query and introduce the notation used later on. We also discuss the concept of the *winnow* operator which is used to derive logical representations of bipolar queries. In Section 3 we compare particular representations under different choices of logical operators, i.e., operators used to model particular logical connectives.

2 The concept of a bipolar query

The very concept of bipolar queries has been introduced by Dubois and Prade [5]. Its basic idea is to distinguish two types of query conditions, which are related to the *negative* and *positive* preferences of a user. The former coincide with the traditional understanding of a condition as a constraint, which defines a set of *feasible* tuples or, equivalently, excludes all tuples that do not satisfy it. The latter, on the other hand, characterizes those tuples which are really desired, with such an un-

ISBN: 978-989-95079-6-8

derstanding that violating such a condition by a tuple does not necessarily exclude it from the consideration. Bipolar queries may be exemplified by the following one:

"Find a *non-expensive* house *and possibly* located *near* a railway station (1)

where the condition referring to the price is a negative one, excluding expensive houses, and the condition referring to the location is a positive one, expressing just a desire to get a house conveniently located, *if possible*.

Now there is a crucial question of the interplay between these two types of conditions which distinguishes different lines of research in this area. Namely, Dubois and Prade basically assume that these conditions should be consistent in such a sense that the set of desired tuples should be a subset of the set of feasible tuples. Then the main question is how to take into account the sets of negative and positive conditions. These are aggregated separately and if the resulting overall negative and positive conditions are not consistent some measures are undertaken to make them so. The answer to such a bipolar query is generated according to the strategy "first select (with respect to the negative condition) and then order (with respect to the positive condition)". This strategy requires a precisiation in case the first condition is fuzzy, i.e., is satisfied by tuples to a degree – what is, of course, the most interesting case, anyway. Then it is not that clear what does it mean to select tuples satisfying a fuzzy condition as they form, in fact, a fuzzy set. Dubois and Prade [6] propose to employ here a lexicographic order of the tuples represented by vectors of the degrees of matching of particular conditions. They propose also a comprehensive representation of bipolar preferences in the framework of the possibility theory [7].

Another line of research explicitly takes into account the conflict between the constraints and the desires and looks for the aggregation of both conditions directly referring to the degree of this conflict. This interpretation is emphasized by the use of the "and possibly" operator in (1). Thus, in general describing a bipolar query we will use the following notation:

$$C \textit{ and possibly } P \tag{2}$$

or, equivalently, an answer to a bipolar query may be defined as the set of tuples: $\{t : C(t) \textit{ and possibly } P(t)\}$. The above form puts emphasis on the question of a proper modeling of the aggregation of both types of conditions, which is expressed here with the use of the "and possibly" operator.

The interest of the database community in this type of queries dates back to the paper by Lacroix and Lavency [8]. They were the first to propose the use of a query comprising two categories of conditions: one which is mandatory (C) and another which expresses just mere preferences (desires) (P). The bipolarity of these conditions becomes evident as soon as one adopts the following interpretation. The former condition C may be seen as expressing the *negative* preferences: the tuples which do not satisfy it are definitely not matching the whole query. The latter condition P, on the other hand, has a *positive* character: a tuple satisfying it is preferred over another tuple not satisfying it, provided both tuples satisfy the mandatory condition C.

For the purposes of a further discussion we will use the following notation. We assume the queries are addressed against a set of tuples $T = \{t_j\}$ comprising a relation. We will identify the negative and positive conditions of a bipolar query with the predicates that represent them and denote them as C and P, respectively. For a tuple $t \in T$, $C(t)$ and $P(t)$ will denote that the tuple t satisfies respective condition (in crisp case) or the degrees of satisfaction, if the conditions are fuzzy. We will also denote the whole bipolar query as (C, P).

According to the original (crisp) approach by Lacroix and Lavency if there are no tuples meeting both conditions then the result of the aggregation is determined by the negative condition C alone. Otherwise the aggregation becomes a regular conjunction of both conditions. Thus the answer to such a query depends not only on the explicit arguments, i.e., $C(t)$ and $P(t)$, but also on the content of the database. This dependence is best expressed by the following logical formula [8]:

$$C(t) \textit{ and possibly } P(t) \equiv C(t) \wedge \exists s(C(s) \wedge P(s)) \Rightarrow P(t) \tag{3}$$

The characteristic feature of such an interpretation of bipolar queries is that if there is no conflict between the conditions P and C, i.e., there are tuples satisfying both of them, then the query turns into a conjunction of the conditions. On the other hand if there are no tuples satisfying both conditions then only condition C is used to select tuples.

Such an aggregation operator has been later proposed independently by Dubois and Prade [9] in the context of default reasoning and by Yager [10, 11] in the context of the multicriteria decision making for the case of so-called *possibilistically qualified criteria*. Yager [11] intuitively characterizes a possibilistically quantified criterion as such which should be satisfied unless it interferes with satisfaction of other criteria. This is in fact the essence of bipolar queries in the sense advocated in this paper. This concept was also applied by Bordogna and Pasi [12] for the textual information retrieval task.

Lacroix and Lavency [8] consider only the case of crisp conditions C and P. Then a bipolar query may be, in fact, processed using the "first select using C then order using P" strategy, i.e., the answer to the bipolar query (C, P) is obtained by, first, finding tuples satisfying C and, second, choosing from among them those satisfying the condition P, if any. If C is crisp and P is fuzzy then the second step consists in non-increasingly *ordering* the tuples satisfying C according to their degree of satisfaction of P. This understanding is predominant in the literature dealing with fuzzy extensions of the original concept of Lacroix and Lavency. Both, direct extensions proposed by Bosc and Pivert [13, 14] as well as sophisticated possibility theory-based interpretation of this concept by Dubois and Prade [6] focus, in fact, on the proper treatment of *multiple* required and preferred conditions, basically assuming the above strategy as the way of combining the negative and positive conditions.

In [15, 16] we propose a "fuzzification" of the formula (3) and study its basic properties. Here we focus on the question of combining fuzzy conditions C and P and will treat them in what follows as atomic. From this point of view it is worth mentioning some other approaches which are of relevance here. Dujmović [17] introduced the concept of the *partial absorption function* which may be used to combine the values of two variables in such a way that one variable controls the influence of the other on the result of their combina-

ISBN: 978-989-95079-6-8

tion. It makes it possible to express the requirement that for a high value of the result a high value of the first variable is mandatory while the high value of the second is desired but not mandatory. When applied to the aggregation of the values of $C(t)$ and $P(t)$ this is somehow similar to the idea of bipolar query expressed by (3), but lacks its dependence on the content of the whole database. This approach may be seen as based on a sophisticated, *dynamic* weighting of the importance of the combined values, where the weights itself depend on the combined values. A similar approach has been proposed by Dubois and Prade; cf., e.g., [4].

The operator "among", very close to the discussed here operator "and possibly", has been proposed by Tudorie [18]. She considers queries of the type "find tuples satisfying a condition P among those satisfying a condition C", which are in fact equivalent to the bipolar queries understood as in (2). The evaluation of a query with the "among" operator is very similar to the one adopted by us for the bipolar query, but is expressed in terms of the rescaling of the linguistic terms used in the condition P. Namely, first the set of tuples satisfying the condition C to a non-zero degree is selected. Then the membership functions of the fuzzy sets representing the linguistic terms appearing in P (such as "near" in (1)) are rescaled taking into account the actual range of the corresponding attributes in the set of tuples selected in the first step. For example, if originally the distance of 2 kilometers from the station has the membership value to the fuzzy set representing the term "near" equals 0.5, and it turns out that it is the shortest distance among the houses selected in the first step (i.e., among non-expensive houses, in case of the query (1)), then this membership degree may be changed to 1 (the actual algorithm of rescaling may, of course, take different forms). Finally, the overall matching degree is computed as a conjunction of the matching degrees against the condition C and the "modified" condition P, i.e., the one for which the rescaled membership function of the linguistic terms is used. Please note that if there is no interference between both conditions (in the sense discussed earlier) then there is no need for rescaling the membership functions and the query turns into a conjunction of both conditions, like in the case of the bipolar query.

Bipolar queries may be also seen a special case of *queries with preferences* proposed recently, for the crisp case, by Chomicki [19]. In the framework of this approach a new relational algebra operator called *winnow* is introduced. This unary operator selects from a set of tuples T those which are *non-dominated* with respect to a given *preference relation* R, $R \subseteq T \times T$. If two tuples $t, s \in T$ are in relation R, i.e., $R(t, s)$, then it is said that the tuple t *dominates* the tuple s with respect to the relation R. Then the *winnow* operator ω_R is defined as follows

$$\omega_R(T) = \{t \in T : \neg\exists_{s \in T}\ R(s, t)\} \quad (4)$$

Thus, for a given set of tuples it yields a subset of the *non-dominated* tuples with respect to R.

The concept of the *winnow* operator may be illustrated with the following example. Let us consider a database of a real-estate agency with a table `HOUSES` describing the details of particular real-estate properties offered by an agency. The schema of the relation `HOUSES` contains, among other, the attributes `city` and `price`. Let us assume that we are interested in the list of the *cheapest* houses in each city. Then the preference relation should be defined as follows

$$R(t, s) \Leftrightarrow (t.\texttt{city} = s.\texttt{city}) \wedge (t.\texttt{price} < s.\texttt{price})$$

where $t.A$ denotes the value of attribute A (e.g., `price`) at a tuple t. Then the *winnow* operator $\omega_R(\texttt{HOUSES})$ will select the houses that are sought. Indeed, according to the definition of the *winnow* operator, we will get as an answer a set of houses, which are non-dominated with respect to R, i.e., for which there is no other house in the same city which has a lower price.

In [16] we proposed a fuzzy counterpart of the *winnow* operator taking into account the fuzziness of the preference relation R and of the related concept of non-dominance as well as the fact that the set of tuples T is also a fuzzy set. It may be expressed as follows:

$$\mu_{\omega_R(T)}(t) = \text{truth}(T(t) \wedge \forall_s\ (T(s) \rightarrow \neg R(s, t))) \quad (5)$$

where $\mu_{\omega_R(T)}(t)$ denotes the value of the membership degree of the tuple t to the fuzzy set of tuples defined by $\omega_R(T)$.

A bipolar query (C, P) may be expressed using the concept of the fuzzy *winnow* operator as follows [19, 16]. Let R be a fuzzy preference relation of the following form (symbols R and P denote both the fuzzy predicates and the membership functions of corresponding fuzzy sets, depending on the context):

$$R(t, s) \Leftrightarrow P(t) \wedge \neg P(s) \quad (6)$$

Then the bipolar query may be expressed as the combination of the selection and fuzzy *winnow* operators $\omega_R(\sigma_C(T))$, i.e.,

$$\mu_{\omega_R(\sigma_C(T))} = \text{truth}(C(t) \wedge \forall_s\ (C(s) \rightarrow (\neg P(s) \vee P(t)))) \quad (7)$$

(where $\sigma_C(T)$ is a usual "fuzzy" extension of the standard relational algebra selection operator, i.e., $\mu_{\sigma_C(T)}(t) = C(t)$).

The definition of the fuzzy *winnow* operator (5) as well as the "fuzzification" of the formula defining a bipolar query (3) leave open the question of a choice of the logical operators which should be used to model particular logical connectives occurring in both formulas. In [16] we show that for a specific choice of them the fuzzy set of tuples obtained using (7) is identical with the fuzzy set defined by (3). In [20] we analyze the properties of the "fuzzified" version of (3) for the broader class of the logical operators. In the next section we further advance this study.

3 Various interpretations of bipolar queries and their properties

In our previous work [15, 16, 20] we studied a specific fuzzy version of the Lacroix and Lavency formula (3) representing an interpretation of the concept of bipolar queries. We have also shown its basic relation with a fuzzy version of the *winnow* operator we proposed. Here we extend this study comparing three formulas that may be used to represent the bipolar query and their properties under different interpretations of the logical connectives occurring in them.

We derive the logical formulas expressing the matching degree of a bipolar query in three different ways (we repeat here some formulas introduced in the previous section for the convenience of the reader):

ISBN: 978-989-95079-6-8

- making a direct "fuzzification" of the formula (3) proposed for the crisp case by Lacroix and Lavency [8]:

$$C(t) \textit{ and possibly } P(t) \equiv C(t) \wedge (\exists s\ (C(s) \wedge P(s)) \Rightarrow P(t)) \quad (8)$$

- making a direct "fuzzification" of the crisp *winnow* operator (4) and applying it with a specific preference relation (6) to obtain a bipolar query representation:

$$C(t) \textit{ and possibly } P(t) \equiv C(t) \wedge \neg\exists s\ ((C(s) \wedge P(s) \wedge \neg P(t))) \quad (9)$$

- using our fuzzy version of the *winnow* operator (5) and applying it as above:

$$C(t) \textit{ and possibly } P(t) \equiv C(t) \wedge \forall s\ (C(s) \Rightarrow (\neg P(s) \vee P(t))) \quad (10)$$

It may be easily seen that in the framework of the classical logic all three above formulas are equivalent. Here we study their properties in case of fuzzy (multivalued) interpretation, in particular taking into account various operators which may be used to model logical connectives. We follow usual approach of modeling conjunction and disjunction by the t-norm and t-conorm operators, respectively [21].

In order to carry out the analysis we consider so-called De Morgan Triples $(\wedge, \vee, \neg)$ that comprise a t-norm operator $\wedge$, a t-conorm operator $\vee$ and a negation operator $\neg$, where $\neg(x \vee y) = \neg x \wedge \neg y$ holds. Three following De Morgan Triples play the most important role in fuzzy logic (cf., e.g., [21] for a justification) $(\wedge_{min}, \vee_{max}, \neg)$, $(\wedge_\Pi, \vee_\Pi, \neg)$, $(\wedge_W, \vee_W, \neg)$, where particular t-norms and t-conorms are defined as follows:

$t-$ *norms*		
$x \wedge_{min} y$	$= \min(x, y)$	*minimum*
$x \wedge_\Pi y$	$= x \cdot y$	*product*
$x \wedge_W y$	$= \max(0, x + y - 1)$	*Łukasiewicz*

$t-$ *conorms*		
$x \vee_{max} y$	$= \max(x, y)$	*maximum*
$x \vee_\Pi y$	$= x + y - x \cdot y$	*probabilistic sum*
$x \vee_W y$	$= \min(1, x + y)$	*Łukasiewicz*

We will refer to these De Morgan Triples in what follows as, respectively, MinMax, Π and W triples. The negation operator $\neg$ in case of all the above De Morgan Triples is defined as: $\neg x = 1 - x$. Both t-norms and t-conorms are by definition associative and thus may be treated as m-ary operators, i.e., expressions like $x \wedge y \wedge \ldots$ and $x \vee y \vee \ldots$ are well defined.

Basically, the general and existential quantifiers are identified in fuzzy logic, for the case of a finite universe, with the maximum and minimum operators, respectively. They may be generalized via the use of other t-norms and t-conorms what leads to the concept of *t-quantifiers* and *s-quantifiers*; cf., e.g., [22]. The truth of a statement involving such a quantifier is computed as follows ($\{a_1, \ldots, a_m\}$ is a finite universe under consideration):

$$\text{truth}(\forall x A(x)) = \mu_A(a_1) \wedge \mu_A(a_2) \wedge \ldots \wedge \mu_A(a_m) \quad (11)$$

$$\text{truth}(\exists x A(x)) = \mu_A(a_1) \vee \mu_A(a_2) \vee \ldots \vee \mu_A(a_m) \quad (12)$$

We use generalized quantifiers while interpreting formulas (8)-(10) and particular t- and s-quantifiers will be denoted in what follows by the $\forall$ and $\exists$ symbol with a subscript indicating underlying t-norm or s-norm, e.g., $\exists_{max}$ denotes a fuzzy existential quantifier which obtains when the t-conorm "maximum" is used.

We consider two implication operators related to a given De Morgan Triple $(\wedge, \vee, \neg)$, so-called S-implications:

$$x \rightarrow_{S-\vee} y = \neg x \vee y \quad (13)$$

and R-implications:

$$x \rightarrow_{R-\wedge} y = \sup\{z : x \wedge z \leq y\} \quad (14)$$

Thus, for particular De Morgan Triples one obtains the following R-implication operators:

$$x \rightarrow_{R-min} y = \begin{cases} 1 & \text{for } x \leq y \\ y & \text{for } x > y \end{cases}$$

$$x \rightarrow_{R-\Pi} y = \begin{cases} 1 & \text{for } x = 0 \\ \min\{1, \frac{y}{x}\} & \text{for } x \neq 0 \end{cases}$$

$$x \rightarrow_{R-W} y = \min(1 - x + y, 1)$$

and the following S-implication operators:

$$x \rightarrow_{S-max} y = \max(1 - x, y)$$

$$x \rightarrow_{S-\Pi} y = 1 - x + x \cdot y$$

$$x \rightarrow_{S-W} y = \min(1 - x + y, 1)$$

Now let us consider the question of the choice of one of the formulas (8)-(10) to represent bipolar queries and an appropriate modeling of the logical connectives occurring therein, i.e., the choice of one of the De Morgan Triples.

In [20] we have shown certain basic properties of the fuzzified version of the original formula (8). Some of them are valid for any choice of logical operators, some are limited to some special cases. We study some of them here again, checking if they are valid also for the formulas (9) and (10). However, first we start with a property identifying the equivalence between formulas (8)-(10) for a certain choice of the logical operators.

Property 1 For a distributive De Morgan triple, i.e., when $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$, and a related S-implication all formulas (8)-(10) are equivalent.

We will show the equivalence between (8) and (9) and the remaining equivalences may be shown in a similar way (the equivalence of (8) and (10) was shown by us in [16] for the specific case of the MinMax De Morgan Triple). Formula (9) may be rewritten as follows:

$$C(t) \wedge \forall s\ (\neg C(s) \vee \neg P(s) \vee P(t))$$

what, due to the assumed distributivity of the triple is equivalent to:

$$C(t) \wedge ((\forall s\ (\neg C(s) \vee \neg P(s))) \vee P(t))$$

and via the following series of transformations leads to (8):

$$C(t) \wedge ((\forall s\ \neg(C(s) \wedge P(s))) \vee P(t)) \equiv$$

$$C(t) \wedge ((\neg\exists s\ (C(s) \wedge P(s))) \vee P(t)) \equiv$$

$$C(t) \wedge (\exists s\ (C(s) \wedge P(s)) \Rightarrow P(t))$$

ISBN: 978-989-95079-6-8

The last equivalence holds for "⇒" being an S-implication.

Among three De Morgan Triples that we consider in this paper only the MinMax triple is distributive. Thus for it and related S-implication we get equivalence of all three formulas (8)-(9). Property 1 shows only a sufficient condition for this equivalence, but for the other two triples some counter-examples for such an equivalence may be easily shown.

In [20] we have shown a property which is worth reminding as it best characterizes the understanding of the bipolar queries adopted here. Namely, if there is no conflict between the required (C) and preferred (P) conditions at all, i.e., there is a tuple fully (to a degree equal 1) satisfying both of them then the formula (8) turns into a regular conjunction of both conditions. This may be formally expressed as follows.

Property 2 If there exists a tuple t such that $C(t) = P(t) = 1$ then a bipolar query (C, P) turns into the conjunction $C \wedge P$.

In [20] we have shown that for any combination of a t-norm, t-conorm and S-implication or R-implication such that $(1 \Rightarrow x) = x$, Property 2 holds in case of the formula (8). Note that the conjunction mentioned in the Property is modeled then via assumed t-norm.

Here, in view of the Property 1, it is clear that in case of the MinMax De Morgan Triple and the S-implication, Property 2 holds also for the formulas (9) and (10). Moreover, it may be easily proved that this property also holds for the MinMax De Morgan Triple and the R-implication in case of the formula (10). Namely, we will show that:

$$\text{truth}(\forall s\ (C(s) \Rightarrow_{R-min} (\neg P(s) \vee_{max} P(t))) = \min_s \begin{cases} 1 & \text{if } C(s) \leq \neg P(s) \vee_{max} P(t) \\ \neg P(s) \vee_{max} P(t) & \text{otherwise} \end{cases} \quad (15)$$

is equal $P(t)$ for any tuple t, under the assumptions of the Property 2. If $P(t) = 1$ then for all tuples s the value of $C(s)$ is lower or equal $\max(1 - P(s), P(t))$ and thus (15) is equal 1, i.e., is equal to $P(t)$. Now let us assume that $P(t) < 1$ and let us denote with u a tuple for which $C(u) = P(u) = 1$ (the existence of such a tuple is assumed in the Property 2). Then, for any tuple t, the minimum over s in (15) is realized for $s = u$ and thus is equal $\max(1 - 1, P(t))$, i.e., $P(t)$, what completes our proof of the Property 2 also for the formula (10) and the R-implication operator.

Some examples may be easily found showing that Property 2 fails for the W and Π De Morgan triples.

Another property of the formula (8), shown in [20], is valid for two other formulas (9) and (10), and may be expressed as follows.

Property 3 If for a tuple t the value of $P(t)$ is equal to 1, then a bipolar query (C, P) turns into $C(x)$.

This property holds for all formulas (8)-(10), for any combination of a t-norm, t-conorm and S-implication or R-implication.

This property is a direct consequence of the general properties of the t-norm ($(x \wedge 0) = 0$), t-conorm ($(x \vee 1) = 1$ and $(0 \vee 0) = 0$) and implication operators ($(x \Rightarrow 1) = 1$).

Now we show some additional properties of the formulas (8)-(10).

Property 4 Let us assume that truth$(\exists s\ (C(s) \wedge P(s))) > 0$. Then, for a De Morgan Triple with a t-norm without zero divisors, i.e., where $\forall x, y > 0\ (x \wedge y) \neq 0$, and the related R-implication, the matching degree computed using (8) for a tuple t fully satisfying the required condition and not satisfying the preferred condition at all (i.e., $C(t) = 1$ and $P(t) = 0$) is equal 0.

This is a property of the R-implication. Notice that this means that such a tuple t will get lower matching degree than a tuple s which satisfies both conditions to a degree ϵ, whatever small ϵ is. Thus it is surely a property we would like to avoid and which makes (8) under both MinMax and Π De Morgan Triples (whose t-norms do not have zero divisors) with related R-implications less appealing as models of the bipolar query.

A similar property, formulated below, is exhibited by (10).

Property 5 Let us assume that there is at least one tuple u such that $C(u) > 0$ and $P(u) = 1$. Then, for a De Morgan Triple with a t-norm without zero divisors and the related R-implication the matching degree computed using (10) for a tuple t fully satisfying the required condition and not satisfying the preferred condition at all (i.e., $C(t) = 1$ and $P(t) = 0$) is equal 0.

Thus also (10) under both MinMax and Π De Morgan Triples with related R-implications is not very appealing as a model of the bipolar query. This property seems to favor the W De Morgan Triple, at least in case of the R-implication and formulas (8) and (10).

Another negative property of (8) for a specific combination of logical operators may be expressed as follows.

Property 6 For the MinMax De Morgan Triple used with related S-implication, the aggregation scheme defined by (8) may lead to the same matching degree for two tuples t and u while t strongly Pareto dominates u, i.e., $C(t) > C(u)$ and $P(t) > P(u)$.

This may be demonstrated with the following example. Let us denote $\exists_s\ (C(s) \wedge P(s))$ with $\exists CP$. Let $\exists CP$ = 0.7 and $C(t) = 1$, $P(t)$ = 0.3, $C(u)$ = 0.3 and $P(u) = 0$. Then the matching degree computed for both tuples is equal 0.3, while t strongly Pareto dominates u.

In fact this property may be supplemented by observing that all tuples t such that $P(t) \leq (1 - \exists CP)$ and $C(t) \geq (1 - \exists CP)$ obtain the same matching degree, equal $1 - \exists CP$. This fact has been observed by Dubois and Prade [5] for a formula similar to (8). However, it should be noted that even for the tuples not satisfying the above conditions, the Pareto domination may be not reflected by (8) used with the logical operators specified by Property 6. For example, for a tuple u, such that $C(u) = P(u) = 0.6$, still assuming $\exists CP = 0.7$, the matching degree is equal 0.6. The same matching degree obtains for t such that $C(t) = 1$ and $P(t) = 0.6$ as well as for t such that $C(t) = 0.6$ and $P(t) = 1.0$, while in both these cases the tuple t Pareto dominates u.

Note, that due to the Property 1, the Property 6 is also valid for two other formulas (9)-(10).

Certainly the list of properties discussed in the paper is not exhaustive and they should be seen as a first attempt at a more comprehensive analysis of the bipolar queries and their various representations proposed here. Such an analysis should provide some hints which representation should be used un-

ISBN: 978-989-95079-6-8

der which conditions and using which set of logical operators. However, already the properties discussed here provide some hints which may be summarized as follows.

We first list some general properties shared by all three formulas (8)-(10) under any combination of the logical operators. These properties may be expressed concisely as the properties of the "and possibly" operator, which is the essence of our understanding of bipolar queries (this operator is denoted below as $\wedge_{possibly}$, but it should be remembered that this operator is not truth-functional):

- monotonicity (but not strict) in both arguments,
- boundary properties: $1 \wedge_{possibly} 1 = 1$ and $x \wedge_{possibly} 1 = x$ (Property 3).

Now let us look at particular formulas (8)-(10) and summarize their properties in combinations with all considered logical operators.

Formula (8) exhibits Property 2 when used with any combination of logical operators. This is surely its advantage as this property is characteristic for our understanding of bipolar queries. Property 4 seems to suggest that the R-implication should be avoided in case of formula (8) (unless it is used in the framework of the W De Morgan Triple, but then both types of implication operators are identical, thus a general hint of avoiding R-implication may be still seen as valid). Property 6 suggests that the MinMax De Morgan Triple is generally not appropriate for the formula (8).

Formula (9) exhibits Property 2 only for the MinMax De Morgan Triple, but Property 6 makes this triple inappropriate to some extent.

Formula (10) also satisfies Property 2 only for the MinMax De Morgan Triple, which is on the other hand somehow inappropriate due to the Properties 5 and 6.

Concluding, if Property 2 is required, what seems to be a reasonable postulate, then the best choice of the representation of the bipolar queries and logical operators to model the logical connectives therein– according to studied properties– seems to be formula (8) with the Π De Morgan Triple and the S-implication operator. Such a choice saves the obtained representation from some negative properties discussed in this paper. Further studies are needed in order to identify a more comprehensive list of postulated properties.

References

[1] P. Bosc and O. Pivert. SQLf: A relational database language for fuzzy querying. *IEEE Transactions on Fuzzy Systems*, 3(1):1–17, 1995.

[2] J. Kacprzyk and S. Zadrożny. Computing with words in intelligent database querying: standalone and internet-based applications. *Information Sciences*, 134(1-4):71–109, 2001.

[3] S. Zadrożny, G. De Tre, R. De Caluwe, and J. Kacprzyk. An overview of fuzzy approaches to flexible database querying. In Galindo [23], pages 34–53.

[4] D. Dubois and H. Prade. Using fuzzy sets in flexible querying: why and how? In T. Andreasen, H. Christiansen, and H.L. Larsen, editors, *Flexible Query Answering Systems*, pages 45–60. Kluwer Academic Publishers, 1997.

[5] D. Dubois and H. Prade. Bipolarity in flexible querying. In T. Andreasen, A. Motro, H. Christiansen, and H.L. Larsen, editors, *Flexible Query Answering Systems, FQAS 2002*, volume 2522 of *Lecture Notes in Computer Science*, pages 174–182. Springer, Berlin, Heidelberg, 2002.

[6] D. Dubois and H. Prade. Handling bipolar queries in fuzzy information processing. In Galindo [23], pages 97–114.

[7] D. Dubois and H. Prade. An overview of the asymmetric bipolar representation of positive and negative information in possibility theory. *Fuzzy Sets and Systems*, 160(10):1355–1366, 2009.

[8] M. Lacroix and P. Lavency. Preferences: Putting more knowledge into queries. In *Proceedings of the 13 International Conference on Very Large Databases*, pages 217–225, Brighton, UK, 1987.

[9] D. Dubois and H. Prade. Default reasoning and possibility theory. *Artificial Intelligence*, 35(2):243–257, 1988.

[10] R.R. Yager. Higher structures in multi-criteria decision making. *International Journal of Man-Machine Studies*, 36:553–570, 1992.

[11] R.R. Yager. Fuzzy logic in the formulation of decision functions from linguistic specifications. *Kybernetes*, 25(4):119–130, 1996.

[12] G. Bordogna and G. Pasi. Linguistic aggregation operators of selection criteria in fuzzy information retrieval. *International Journal of Intelligent Systems*, 10(2):233–248, 1995.

[13] P. Bosc and O. Pivert. Discriminated answers and databases: fuzzy sets as a unifying expression means. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 745–752, San Diego, USA, 1992.

[14] P. Bosc and O. Pivert. An approach for a hierarchical aggregation of fuzzy predicates. In *Proceedings of the Second IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'93)*, pages 1231–1236, San Francisco, USA, 1993.

[15] S. Zadrożny. Bipolar queries revisited. In V. Torra, Y. Narukawa, and S. Miyamoto, editors, *Modelling Decisions for Artificial Intelligence (MDAI 2005)*, volume 3558 of *LNAI*, pages 387–398. Springer-Verlag, Berlin, Heidelberg, 2005.

[16] S. Zadrożny and J. Kacprzyk. Bipolar queries and queries with preferences. In *Proceedings of the 17th International Conference on Database and Expert Systems Applications (DEXA'06)*, pages 415–419, Krakow, Poland, 2006. IEEE Comp. Soc.

[17] J.J. Dujmović. Partial absorption function. *Journal of teh University of Belgrade, EE Dept.*, 659:156–163, 1979.

[18] C. Tudorie. Qualifying objects in classical relational database querying. In Galindo [23], pages 218–245.

[19] J. Chomicki. Querying with intrinsic preferences. *Lecture Notes in Computer Science*, 2287:34–51, 2002.

[20] S. Zadrożny and J. Kacprzyk. Bipolar queries using various interpretations of logical connectives. In *Foundations of Fuzzy Logic and Soft Computing*, Lecture Notes in Computer Science, pages 181–190. Springer, 2007.

[21] J. Fodor and M. Roubens. *Fuzzy Preference Modelling and Multicriteria Decision Support*. Series D: System Theory, Knowledge Engineering and Problem Solving. Kluwer Academic Publishers, 1994.

[22] R. Mesiar and H. Thiele. On T-Quantifiers and S-Quantifiers. In V. Novak and I. Perfilieva, editors, *Discovering the World with Fuzzy Logic*, pages 310–326. Physica-Verlag, Heidelberg New York, 2000.

[23] J. Galindo, editor. *Handbook of Research on Fuzzy Information Processing in Databases*. Information Science Reference, New York, USA, 2008.

ISBN: 978-989-95079-6-8