# Flexible queries in relational databases
## —The example of the division operator—

Patrick Bosc
I.R.I.S.A./E.N.S.S.A.T., Lannion, France

Didier Dubois
I.R.I.T., Université Paul Sabatier, Toulouse, France

Olivier Pivert
I.R.I.S.A./E.N.S.S.A.T., Lannion, France

and Henri Prade
I.R.I.T., Université Paul Sabatier, Toulouse, France

*Abstract*

Allowing for flexible queries enables database users to express preferences inside minimal requirements, and, if necessary, priorities inside compound queries. In other words, clear-cut properties can be refined by ordering the interpretations compatible with them, according to user's preferences. Often the representation of these preferences can be viewed as modelling linguistic-like terms in requests. In this paper, the theoretical issues raised by the introduction of flexible queries are studied in the case of the division operator, in the framework of fuzzy sets and possibility theory. The notion of division is well-known in the context of regular relations and the extension of this operation to fuzzy relations (induced by the flexible queries) is investigated. Several types of extended divisions can be envisaged, depending on the meaning of the grades attached to the tuples of the fuzzy relations (degree of fulfillment of gradual properties, level of importance of components in a query, or uncertainty pervading data). We focus on the first two meanings which are associated with different multiple-valued logic implications and we examine their properties and their expression in the framework of an extended SQL-like language where no specific construct for the division is available.

## 1. Introduction

Database management systems are software components designed to store, retrieve, update and control large amounts of permanent data. Up to now, they have been mainly used for business purposes so as to respond to a certain range of applications where data are precisely known (no imprecision, no uncertainty) and retrieval procedures are intended to get data according to their compliance with crisp conditions (whose satisfaction is not a matter of

*Correspondence to*: P. Bosc, I.R.I.S.A./E.N.S.S.A.T., Université de Rennes I, 6 rue de Kérampont, B.P. 447, 22305 Lannion Cedex, France, Emails: {bosc, pivert}@merlin.enssat.fr; {dubois, prade}@irit.irit.fr

degree). The last few years have witnessed a tremendous increase in the use of computers in more and more domains, the need for managing new kinds of data and for providing new capabilities for storage, access and display. In this respect, one may imagine introducing what is often dubbed "uncertainty" into databases. This term may refer to two main streams of problems. On the one hand, one wants to store and manipulate incomplete data (i.e., the available information about attribute values may be tainted with imprecision and/or uncertainty for some items). This can be achieved through different formalisms (probability [2], [23] and possibility theory [19], [20], [22], in particular), but in that case, the retrieval process will also return results involving some uncertainty (if we are uncertain about the precise value of John's age, we cannot always be sure that John does (or does not) satisfy a given requirement in the context of a query selecting people on basis of their age). On the other hand, the term "uncertainty" is sometimes used for referring to flexible queries since then one may consider that there is some ambiguity pertaining to their meaning. In fact, flexible queries are useful for describing preferences and thus for getting an ordered set of answers accordingly.

This paper deals with this second type of "uncertainty" and is concerned essentially with database language extensions in order to deal with more expressive requirements. Indeed, consider a query such that, for instance, "retrieve the apartments which are *not too expensive* and *not too far* from downtown". In such a case, there does not exist a definite threshold for which the price becomes suddenly too high, but rather we have to discriminate between prices which are perfectly acceptable for the user, and other prices, somewhat higher, which are still more or less acceptable (especially if the apartment is close to downtown). Note that the meaning of vague predicate expressions like "not too expensive" is context/user dependent, rather than universal. Fuzzy set membership functions [26] are convenient tools for modelling user's preference profiles and the large panoply of fuzzy set connectives can capture the different user attitudes concerning the way the different criteria present in his/her query compensate or not; see [4] for a unified presentation in the fuzzy set framework of the existing proposals for handling flexible queries. Moreover in a given query, some part of the request may be less important to fulfill (e.g., in the above example, the price requirement may be judged more important than the distance to downtown); the handling of importance leads to the need for weighted connectives, as it will be seen in the following.

Thus, flexible queries are motivated by the expression of preferences, and of relative levels of importance. However the use of queries involving fuzzily bounded categories may be also due to an interest for more robust evaluations. This is the case in a query like "find the average salary of the *young* people stored in the database", where the use of a predicate like "young" (whose meaning is clearly context-dependent) does not here refer to the expression of a preference, but is rather a matter of convenience since the user is not obliged to set the boundaries of the category of interest in a precise and thus rather arbitrary way; in such a case, a range of possible values for the average salary instead of a precise number will be returned to the user. This range can be viewed as bounded by the lower and the upper expected values of a fuzzy number; see [12].

Some works already exist, that extend the relational model of data [8] so as to deal with ill-known data, and/or the relational algebra so as to allow for imprecise queries [6], [19], [20], [21]. Thus, extensions of selection, projection, Cartesian product, join, and set operations have been deeply investigated by these authors and others in order to handle preference levels. However, the operator known as "relational division" has received little attention in that perspective, perhaps because of its non primitivity, but also because its extension involves a more intricate treatment of the preference levels. In this paper division is considered in a relational framework although the notion of division is not strictly tied to relations and could also apply to object-oriented databases.

If, for example, we know the prices of products (contained in relation S) on the one hand and the quantities ordered by different stores (forming relation R) on the other hand, the query looking for the stores having ordered at least 10 pieces of all products over $15, is a matter of division (namely: R' divided by S', where R' and S' are the sub-relations of R and S pertaining to names of products and selected prices, and to names of stores, names of products and selected quantities respectively). At this point, one may imagine changing the previous query to introduce some graduality or flexibility. A first query would be: "find the stores having ordered a *moderate* number of pieces of all *medium-priced* products", which, intuitively should call on a division involving gradual (fuzzy) relations, R" (resp. S") expressing the extent to which the number of pieces ordered is moderate (resp. the price of a product is medium). This query may have (at least) two different understandings depending on the nature of the interaction between the price of products (with respect to medium) and the number of pieces ordered (with respect to moderate). One interpretation is: "the closer to medium the price of a product is, the more moderate the number of ordered pieces should be". In other words, we are only interested in retrieving a store insofar as its ordered quantities of medium-priced products are moderate, with the understanding that the level of moderateness of the ordered quantities is preferably greater than the level of medium-pricedness for *all* products (assuming commensurate scales for these levels).

Another slightly different understanding is: "retrieve the stores having ordered a *moderate* quantity of all the *important* products", with the understanding that the idea of importance is graded. Keeping in mind the example with which we start, the level of importance is supposed here to directly reflect the level of medium-pricedness of the product. The idea here is that, on the one hand, a retrieved store is totally satisfactory only if all somewhat important products are ordered in quantities which are fully moderate; on the other hand, a store is completely rejected only if there exists a fully important product whose ordered quantity is not at all moderate. Note that a store cannot be fully rejected here on the basis of an ordered quantity which is not at all moderate if the associated product is not fully important (here, its price is not really medium).

An additional degree of flexibility would be offered by weakening the universal quantifier "all" into "most", as exemplified by the request: "retrieve the stores having ordered a *moderate* number of pieces of *most* products". Such a request can be treated by adapting the latter interpretation in the following way (as suggested in Section 3): the products are decreasingly ranked according to their level of moderateness (with respect to ordered quantities) and most of the products in this ordered list starting with the first ones are considered as important. Lastly, we may imagine more sophisticated queries where the scope of '*most*' focuses on a fuzzy class of products, like in the query "retrieve the stores having ordered a *moderate* number of pieces of *most medium-priced* products". Note also that apart from queries involving different relaxations of the universal quantifier, and requiring extended division, we may also deal in the fuzzy set framework with queries like "retrieve the stores which have ordered a *moderate* quantity of *at least one medium-priced* product" which calls on selections and joins of fuzzy relations.

In this paper, we are mainly concerned with two extensions of the division, corresponding to the two first above interpretations involving levels of fulfillment of properties and levels of importance. This work takes place in the context of regular relational databases, whose contents is not pervaded with uncertainty, and to which flexible queries, returning more or less satisfactory items, are addressed. It deals more particularly with the division of fuzzy relations, the possible meanings of this operation, its properties and then its expression in an extended relational algebra as well as in an SQL-like query language. The paper is organized as follows. In Section 2, the definition of the relational division is recalled along with some of its properties and how to express it in SQL. After a brief review of previous works related to the extension of the division in the presence of fuzzy relations, possible semantics for the division of fuzzy relations are proposed in Section 3 in terms of the choice of the appropriate fuzzy implication underlying the definition of the fuzzy division. The properties and associated expressions in SQLf (an extended version of SQL supporting fuzzy queries) are described in Section 4.

## 2. The Relational Division

In the framework of the relational model of data, a universe is modelled as a set of relations (in a mathematical sense, i.e., a relation R is a subset of the Cartesian product of some domains) which can be manipulated with the help of specific operators known as the relational algebra. Let us denote R (resp. S) a relation defined on the set of attributes X (resp. Y). X also denotes the relation made of all the tuples made of values in the domains (restricted to the values present in the database) of attributes in X. The most usual operators in the relational algebra are:

- union (when X = Y), $R \cup S = \{t \mid t \in R \lor t \in S\}$,
- intersection (when X = Y), $R \cap S = \{t \mid t \in R \land t \in S\}$,
- difference (when X = Y), $R - S = \{t \mid t \in R \land t \notin S\}$,
- Cartesian product, $R \times S = \{(r,s) \mid r \in R \land s \in S\}$,
- selection, $R{:}\varphi = \{r \mid r \in R \land \varphi(r)\}$ ($\varphi$ is a Boolean predicate),
- projection, $R[A] = \{u \mid u \in A \land \exists v \in (X - A) \land (u,v) \in R\}$ (A is a subset of X),
- join, $R[A\ \theta\ B]S = \{(r,s) \mid r \in R \land s \in S \land (r[A]\ \theta\ s[B])\}$ where A (resp. B) is a subset of X (resp. Y), $\theta$ is a comparison operator, and r[A] stands for the A-value in tuple r.

Let R(A,X) denote a relation R such that a tuple in R is made of a subtuple of values of attributes in A followed by a subtuple of values of attributes in X. Beyond these operations, the division of R(A,X) by S(A,Y) denoted by $R[A \div A]S$, where A is a subset of attributes common to R and S, aims at determining the X-values connected in R with *all* the A-values appearing in S. Formally, this operation can be defined in several ways. Viewing the relation R as inducing a multiple-valued mapping $\Gamma$ which associates to $a \in A$ the set $\Gamma(a) = \{x \in X \mid (x,a) \in R\}$, $R[A \div A]S$ is nothing but the *lower image* of S[A] by $\Gamma$, namely the set $\{x \in X \mid \forall a \in S[A], (x,a) \in R\}$. The upper image would be obtained by changing '$\forall$' into '$\exists$' in this definition. Then, the quotient operation $R[A \div A]S$ can be defined under one of the equivalent forms:

– $x \in R[A \div A]S$ if and only if $\forall a \in S[A], (x,a) \in R$ (1)
– $x \in R[A \div A]S$ if and only if $a \in S[A] \Rightarrow (x,a) \in R$ (2)
– $x \in R[A \div A]S$ if and only if $S[A] \subseteq \Gamma^{-1}(x)$ where $\Gamma^{-1}(x) = \{a \in A \mid (x,a) \in R\}$ (3)
– $R[A \div A]S = \bigcap_{a \in S[A]} \Gamma(a)$ (4)

Note that the division operator does not really make sense when $S = \emptyset$ since then any tuple in R[X] will be obtained. It can also be defined in terms of other relational operators (this shows its non primitivity) noticing, due to (1) that the division comes down to discarding from R[X] all x ∈ R[X] such that ∃ a ∈ S(A), (x,a) ∉ R. It gives

$$R[A \div A]S = R[X] - (R[X] \times S[A] - R)[X]. \quad (5)$$

In this formula, the expression (R[X] × S[A] – R) determines the tuples associating values of attributes X and A that are missing in R and then, X-values present in this set must be discarded from the final result, which is done by the outermost difference.

Coming back to the example given in the introduction, the relations PRODUCT(p#, price) and ORDERS(store, p#, quantity) are assumed to be available and the query: "find the stores which have ordered at least 10 pieces of all products over $15" can be expressed: O[p#÷p#]P with O = (ORDERS: quantity > 10 [store, p#]) and P = (PRODUCT: price ≥ 15). If we take the extensions:

ORDERS

| store | p# | qty |
|---|---|---|
| s2 | p2 | 25 |
| s2 | p3 | 20 |
| s2 | p4 | 20 |
| s1 | p1 | 25 |
| s1 | p4 | 4 |
| s1 | p3 | 12 |
| s1 | p2 | 30 |
| s3 | p4 | 12 |
| s3 | p2 | 14 |

PRODUCTS

| p# | price |
|---|---|
| p4 | 20 |
| p1 | 10 |
| p3 | 18 |
| p2 | 25 |

for the relations ORDERS and PRODUCTS, we get the intermediate relations via selection:

O

| store | p# |
|---|---|
| s2 | p2 |
| s2 | p3 |
| s2 | p4 |
| s1 | p1 |
| s1 | p3 |
| s1 | p2 |
| s3 | p4 |
| s3 | p2 |

P

| p# | price |
|---|---|
| p4 | 20 |
| p3 | 18 |
| p2 | 25 |

and the result contains a single element (s2). It is important to notice that in O, the projection onto the pair of attributes (store, p#) is mandatory, otherwise the answer obtained would correspond to the stores which have ordered a given quantity (≥ 10) of products over $15 (since what would be retrieved is pairs (<store>, <quantity>)).

The expression of the division in the SQL language is not quite straightforward since no specific syntactic feature is provided. If we assume for the sake of simplicity that both A and X are atomic attributes, an initial way of expressing a division stems from expression (5):

*select distinct* X *from* R R1 *where not exists*
  (*select* * *from* S *where* A *not in*
    (*select* A *from* R R2 *where* R2.X = R1.X)) (6)

where R1 and R2 are variables renaming the (two) instances of relation R used in the query, R.X denotes the X-component of relation R and * is a shorthand for the set of all the attributes of the concerned relation(s). From a semantic point of view, in this expression, an X-value is selected if there is no A-value in S which is not connected with this X-value in R. Another approach is to start with expression (3) and to use the partitioning mechanism available in SQL to group the tuples of R according to their X-value (use of the key-word "group by"). A partition is selected (thanks to a set condition introduced by the key-word "having") if its set of A-values includes at least the A-values present in S, which leads to:

*select* X *from* R *group by* X *having set*(A) *contains* (*select* A *from* S). (7)

Unfortunately, if this purely set-oriented formulation was possible in SEQUEL2 [7], it has not been kept in the SQL norm and consequently, it is necessary to transform it. To do so, Kim [17] has pointed out an alternative (also based on set comparisons). An X-value x is selected if the subset of A-values tied to x and belonging to S equals the set of A-values in S:

*select distinct* X *from* R R1 *where*
  (*select* A *from* R R2, S *where* R2.X = R1.X *and* R2.A = S.A) = (*select* A *from* S). (8)

But, once again, this construct is not quite standard since it calls on a set comparison (equality). Finally, a formulation supported by SQL and close to the previous two, is based on a comparison of cardinalities according to the expression:

*select* X *from* R *where* A *in* (*select* A *from* S)
*group by* X *having count*(*) = *select count*(*distinct* A) *from* S (9)

where, here again, the star symbol replaces the set of all the attributes of R.

## 3. Semantics for the Division of Fuzzy Relations

### *3.1. Previous Works*

The need for generalizing the relational division so as to take into account some lexical imprecision has been felt by several researchers according to two main directions. The first one aims at defining the division of two fuzzy relations [18], [21], whereas the other is more interested in relaxing the universal quantification present in a division. This latter approach, proposed by Yager [25], introduces a fuzzy quotient operator which involves fuzzy quantifiers like "almost all" or "most" which relaxes the universal quantifier. Yager suggests defining the membership grade of an element x with respect to the result of the fuzzy division as the degree of truth of the proposition "for almost all elements a in S[A], (x, a) is in R", representing "almost all" through an ordered weighted average operator [24].

In none of these approaches, do the authors clearly show how their definition generalizes the usual one and no interpretation or justification (intuitive or formal) of the final grades resulting from the calculus is provided. In particular, the interaction between the degree of an element in S ($\mu_S(s)$) and that of the related tuple in R ($\mu_R(x, s[A])$) is modeled either as a conjunction [18], or an implication [21], [25]. Moreover, the properties of the chosen definition are not exhibited. That is precisely the goal of this paper to investigate more precisely semantic issues connected to the division of two fuzzy relations.

### *3.2. Division and Fuzzy Implications*

Since the division operator corresponds to the definition of a lower image in the classical case, it seems natural to envisage fuzzy division operators as lower images of fuzzy mappings. Lower (and upper) images have been extended to fuzzy relations [14]. Let us assume now that R and S are fuzzy relations, i.e., that their tuples are weighted by a number between 0 and 1. Then, a natural extension of the division, suggested in [15], stems from (1)-(2) where the usual implication is changed into some multiple-valued one and the universal quantifier is interpreted as a generalized conjunction:

$$\mu_{R[A \div A]S}(x) = \min_s \mu_S(s) \rightarrow \mu_R(x,s[A]) \tag{10}$$

where $\mu_R(r)$ denotes the weight associated with the tuple r in R and s[A] is a subtuple restricted to the attributes in A. (10) is one of the fuzzy relational products introduced in [1] whose properties have been more recently studied in [9]. It is also the generalization of the inclusion given in (3) and in this respect, the extension considered here is still a set-oriented operation. There are three main families of fuzzy implication connectives [13]: the S-implications, the R-implications and the reciprocals of R-implications. S-implications are of the form $a \to b = n(a * n(b))$ where n is an involutive order-reversing, negation operation, and $*$ is a conjunction operation modelled by a triangular norm. R-implications are obtained by residuation, namely $a \to b = \sup\{c \in [0,1], a * c \leq b\}$. While $a \to b = n(b) \to n(a)$ holds for S-implications, it is not generally the case for R-implications where the reciprocal $n(b) \to n(a)$ gives birth to a third kind of implications. In an ordinal setting where we use $*$ = min and the order-reversing operation $n(a) = 1 - a$, we respectively get Dienes implication $a \to b = \max(1 - a, b)$ as a S-implication, Gödel implication ($a \to b = 1$ if $a \leq b$, $a \to b = b$ if $a > b$) as a R-implication, and its reciprocal $n(b) \to n(a) = 1$ if $a \leq b$, $n(b) \to n(a) = n(a) = 1 - a$ if $a > b$.

It is worth noticing that the use of min and max operations, of an involutive negation, and of the associated R and S implications only requires a totally ordered scale. So the use of [0,1] as a scale is not compulsory. An ordinal scale made of a finite number of levels can be enough. Then the order-reversing operation is simply defined by reversing the scale. Thus, by ordinal scale we mean a scale where only the ordering between the levels is meaningful. In the following, we mainly consider operations defined from the min and the order-reversing operation $n(a) = 1 - a$, keeping the scale [0,1] for the sake of simplicity, but using [0,1] as an ordinal scale only. This view is not much demanding and makes easier the elicitation of the membership degrees. When operations other than min and max are meaningful in practice on a scale such that [0,1], we may take $*$ as the product, or use $a * b = \max(0, a + b - 1)$; in this latter case the three above types of implications reduce to Lukasiewicz implication ($a \to b = \min(1, 1 - a + b)$).

### *3.3. Possible Meanings of Graded Tuples*

The proper choice of the implication in (10) depends on the intended meaning of the weights. Three possible meanings of the grades can be distinguished [15]:

a) fulfillment: the weight expresses to what extent a gradual property is fulfilled, e.g., the extent to which a quantity is moderate or a price is medium in our introductory example,

b) importance: the weight expresses importance, e.g., we are looking for stores having ordered products p1, p4 and p5, but for instance it is more important to order p4 than p1 (i.e., $\mu_{PR}(p4) > \mu_{PR}(p1)$ where PR denotes the fuzzy set of preferred products), which is specially of interest to express if the database would contain no store having ordered both p1 and p4 (then we shall be glad to retrieve stores having ordered at least p4 which is more important); in other words, the idea, in the case of the division operation is to weaken the requirement for the subtuple x to be in relation R with *all* the elements in S[A], into the requirement of being in relation R with the *most important* elements in S[A], where the idea of importance is graded, i.e., some are more prioritary than others,

c) (un)certainty: the weight expresses (un)certainty, e.g., $\mu_{married}$(John, Jill) = 0.8 then may express that we are not completely sure that John is married with Jill. The marriage between two persons is clearly an all-or-nothing matter which is true or not, and 0.8 is a certainty degree. This degree of certainty can be estimated by a probability degree or by a more qualitative necessity degree [11]. This third interpretation of the grade which should not be confused with the two others (see [16] for a discussion) is not considered in this paper which focuses on flexible queries rather than on uncertain data.

### *3.4. Two Interpretations for the Extended Division*

In the remainder of this paper, we place ourselves in the context of regular databases and we only consider fuzzy relations issued from regular ones by means of a selection on the basis of fuzzy predicates, or by means of the direct assessment of levels of importance. Consequently, only situations (a) and (b) may happen. It is clear that regarding the division R[A ÷ A]S, $\mu_S(s)$ may be interpreted either as a degree of fulfillment (i.e., a kind of threshold or objective that should be attained) or as a degree of importance of the considered value s[A]. Besides, $\mu_R(x,s[A])$ may only be a degree of fulfillment, but not a degree of importance (since degrees of importance are linked to a relaxation of the universal quantifier which bears on a subset of S). It is why we are going to consider the two following cases:

| | $\mu_R(x,s[A])$ | $\mu_S(s)$ |
|---|---|---|
| case 1 | degree of fulfillment | degree of fulfillment |
| case 2 | degree of fulfillment | degree of importance |

**Case 1.** When dealing with degrees of fulfillment, it seems natural to consider that a tuple completely satisfies the query, i.e., belongs to R ÷ S with the degree 1 as soon as the degree of required fulfillment (playing the role of a threshold) $\mu_S(s)$ is less or equal to the degree of fulfillment $\mu_R(x,s[A])$ of the stored item for each s of S. Formally, we should have:

$$\mu_{R[A \div A]S}(x) = 1 \Leftrightarrow \forall s,\ \mu_S(s) \leq \mu_R(x, s[A]).$$

Otherwise, $\exists$s, $\mu_S(s) > \mu_R(x,s[A])$, i.e., the tuple (x, s[A]), satisfies the property to a degree less than the required one. Then, the most natural proposal is to keep for x (w.r.t. to a given s) the same weight as for (x, s[A]), which leads to $\mu_S(s) \rightarrow \mu_R(x, s[A]) = \mu_R(x,s[A])$ and to $\mu_{R[A \div A]S}(x) = \min_{s:\mu_S(s)>\mu_R(x,s[A])} \mu_R(x, s[A])$ in this case. Indeed the only other alternative (with an ordinal scale) would be the contraposition of Gödel implication which is also such that $\mu_{R[A \div A]S}(x) = 1$ if and only if $\forall$s, $\mu_S(s) \leq \mu_R(x, s[A])$, but that leads to $\mu_S(s) \rightarrow \mu_R(x, s[A]) = 1 - \mu_S(s)$ if $\mu_S(s) > \mu_R(x, s[A])$. But this would be strange: for instance if $\exists$s, $\mu_S(s) = 1$ (i.e., s is a fully medium-priced product in our example) and $\mu_R(x, s[A]) = 0.9$ (i.e., the ordered quantity is almost moderate) it would not be satisfactory to claim that $\mu_S(s) \rightarrow \mu_R(x, s[A]) = 0$ (leading to $\mu_{R[A \div A]S}(x) = 0$), instead of 0.9 (with Gödel implication). Gödel implication is faithful of the level of fulfillment $\mu_R(x, s[A])$ which is effectively reached (although it is less than $\mu_S(s)$), while its reciprocal keeps memory of the required level of fulfillment ($\mu_S(s)$) when this level is not reached. Thus, we use Gödel implication in (10): $a \rightarrow_G b = 1$ if $a \leq b$, $a \rightarrow_G b = b$ otherwise, and we adopt the definition:

$$\mu_{R[A \div A]S}(x) = \min_s (\mu_S(s) \rightarrow_G \mu_R(x,s[A])). \tag{11}$$

The use of Gödel implication agrees with the idea that x will be retrieved with a degree which is all the higher as $\mu_R(x, s[A])$ is large when $\mu_S(s)$ is large, for all s. In our introductory examples it means that the more medium-priced the products ordered by the store are, the more moderate the number of pieces ordered should be, for the retrieved stores. With Gödel implication, when $\mu_R(x, s[A])$ is less than $\mu_S(s)$, the result of the implication is $\mu_R(x, s[A])$ and this evaluation is *absolute* since the result does not depend at all on $\mu_S(s)$. Sometimes, if we are no longer on a purely ordinal scale, a more convenient behavior could be a result that reflects the *ratio* $\mu_R(x, s[A]) / \mu_S(s)$. Provided that the *relative* level of satisfaction of the property by the tuple in R with respect to the level required in S is meaningful for the scale, we can then use Goguen implication, namely a R-implication generated by $*$ = product and defined by $a \rightarrow b = 1$ if $a = 0$, $a \rightarrow b = \min(1, b / a)$ if $a \neq 0$. Let us illustrate this with an example where R and S have the following extensions:

R

| X | A | μ |
|---|---|---|
| x1 | a1 | 1 |
| x1 | a2 | .3 |
| x2 | a2 | .2 |
| x2 | a1 | 1 |

S

| A | μ |
|---|---|
| a1 | .6 |
| a2 | .4 |

The result of the division according to (11) yields {x1 / .3, x2 / .2}, whereas it is {x1 / .75, x2 / .5} with the ratio point of view (Goguen implication), where the subtuple is followed after the '/' by its degree of membership to the result of the division. It is important to notice that, according to these two implications, any element somewhat present in S is mandatory in R (if $\mu_S(a) > 0$ and $\mu_R(x, a) = 0$, $\mu_{R \div S}(x) = 0$). This behavior is obviated by Lukasiewicz implication whose result would be $1 - \mu_S(a)$ in this case. This would be strange since $(x,a) \notin R$ when $\mu_R(x,a) = 0$. Indeed viewing (10) as computing a degree of inclusion between fuzzy sets, Lukasiewicz implication yields a non-zero degree of inclusion of S into R, even if R is empty provided that S is not normalized ($\nexists s, \mu_S(s) = 1$). Moreover, Lukasiewicz implication requires more than an ordinal scale, and besides, it does not enable us to distinguish between the Case 1 and Case 2 interpretations. In the rest of the paper, we consider Gödel implication only, when dealing with Case 1 interpretation, since it requires an ordinal scale only.

**Case 2.** In this case, the complete satisfaction of the query by a tuple seems to demand that all the requirements in S, whatever their importance, be included in the set of tuples with maximum fulfillment, i.e.:

$$\mu_{R \div S}(x) = 1 \Leftrightarrow (\forall s, \mu_S(s) > 0 \Rightarrow \mu_R(x, s[A]) = 1).$$

Moreover, it seems natural that $\mu_{R \div S}(x) = 0$ *only if* for at least one s of S *both* $\mu_S(s) = 1$ (the requirement has the maximum level of importance) and $\mu_R(x,s[A]) = 0$ (the tuple does not at all fulfill the requirement). But, $\mu_{R \div S}(x)$ is allowed to be strictly positive if $\mu_R(x, s[A]) = 0$ provided that $\mu_S(s) < 1$, i.e., the requirement is not completely important (therefore, the requirement can be forgotten to some extent and a minimal level of satisfaction is guaranteed which should be all the greater than the level of importance is smaller). This desired behavior leads to define the quotient operation for this case by using Dienes implication $a \rightarrow_D b = \max(1 - a, b)$, and to adopt:

$$\begin{aligned} \mu_{R[A \div A]S}(x) &= \min_s (\mu_S(s) \rightarrow_D \mu_R(x, s[A])) \\ &= \min_s \max(1 - \mu_S(s), \mu_R(x, s[A])) \qquad (12) \end{aligned}$$

where S is a normalized fuzzy relation ($\exists s_0, \mu_S(s_0) = 1$) in order to have an appropriate scaling of the levels of importance. In practice, if S is not normalized, e.g., there is no fully medium-priced product in the example of the introduction, it means that the query calls for a division by a subset which is somewhat empty (since there is no element with membership 1) and then the user should be notified and should reformulate the query.

For the sake of simplicity we have used the same example (about medium-priced products and moderate ordered quantities) for illustrating the two interpretations we have discussed here. However, with this particular example, the interpretation where both $\mu_R$ and $\mu_S$ are interpreted in terms of degrees of fulfillment (and which leads to the use of Gödel interpretation) may be felt more natural. If the query is modified into "find the stores having ordered a moderate number of pieces of the important (or the prioritary) products", where importance or priority is a matter of degree, the Dienes implication-based interpretation is more natural. Note that in such a case the degrees $\mu_S(s)$'s are no longer obtained through the evaluation of a fuzzy condition (like "the product is medium-priced"), but directly through the assessment of the level of importance or priority of each product which should be either given in the query, or defined once for all by the user. Clearly, this type of assessment is preferably often done on a purely ordinal scale with a finite number of levels, which is compatible with the representation we use. Moreover since the degrees $\mu_S(s)$'s are then directly assessed, it makes the control of the normalization of S easier.

The expression (12) can be viewed as a weighted conjunction, i.e., a conjunction of the degrees of fulfillment $\mu_R(x, s[A])$ weighted by the levels of importance $\mu_S(s)$ (see [10] for an introduction to weighted min and max operations). Indeed, if all the requirements are equally important, i.e., $\forall s$, $\mu_S(s) = 1$, (12) reduces to $\mu_{R[A \div A]S}(x) = \min_s \mu_R(x, s[A])$; if $\exists s_0$, $\mu_S(s_0) = 0$ ($s_0$ has no importance), the corresponding $\mu_A(x, s_0[A])$ is not taken into account in (12).

Using *ordered* weighted conjunctions [15], we can estimate to what extent x is such that $(x,a) \in R$ holds for *most* tuples a in S[A] (or any other quantifier modelled by a fuzzy set), rather than for all the important tuples a as in (12) (we only consider the case where S[A] is not a fuzzy set in the following). This can be done by i) rank-ordering the k subtuples s in S according to the decreasing values of $\mu_R(x, s[A])$ for $s[A] \in S[A]$, i.e., $\mu_R(x, s_{\sigma(1)}[A]) \geq \ldots \geq \mu_R(x, s_{\sigma(i)}[A]) \geq \ldots \geq \mu_R(x, s_{\sigma(k)}[A])$ where $\sigma$ denotes a permutation, and by weighting the $\mu_R(x, s_{\sigma(i)}[A])$'s by $\mu_I(i)$ (in place of $\mu_S(s)$ in (12)) with $\mu_I(1) = 1$ and $\mu_I(i) \geq \mu_I(i+1)$ in such a way that the weight of importance $\mu_I(i)$ is all the greater as $\mu_R(x, s_{\sigma(i)}[A])$ is larger. Moreover $1 - \mu_I$, is the membership function of a fuzzy set, defined on $\{1, \ldots, k\}$, of integers close to k, and thus provides here a representation of the idea of "most". Thus if $S[A] = \{a_1, \ldots, a_{10}\}$, i.e., $k = 10$, we rank-order all the tuples $(x, a_i)$ in R where $1 \leq i \leq 10$, according to the decreasing values of the $\mu_R(x, a_i)$'s. Assume our idea of 'most' is here modelled by $\mu_{most} = 1 - \mu_I$, with $\mu_{most}(10) = 1$, $\mu_{most}(9) = 0.8$, $\mu_{most}(8) = 0.5$, $\mu_{most}(x) = 0$ for $x = 1, 7$ (in general 'most' will be defined in terms of relative cardinality w.r.t. to the cardinality of S[A]). Then the tuples $(x, a_i)$ in R with $a_i \in S[A]$ with the higher degrees of fulfillment are associated with the higher levels of importance. Note that some levels of importance are now zero since we are interested in *most* elements in S[A] but not in *all*.

Thus, the requirement that "$(x,a) \in R$ holds for most tuples a in S[A]" is understood as the requirement that "if tuple a is important then $(x,a) \in R$" (where tuple a is all the more important as (x,a) has a high level of membership to R). See [15] for details. The comparison of this approach with Yager [25]'s ordered weighted average-based proposal is left for further research.

### *3.5. An Example*

Let us consider the two following fuzzy relations:

R

| X | A | μ |
|---|---|---|
| x1 | a1 | .8 |
| x1 | a2 | .7 |
| x1 | a3 | 1 |
| x2 | a1 | .5 |

S

| A | μ |
|---|---|
| a1 | 1 |
| a2 | .4 |

This formal example corresponds to the introductory one if we consider that X represents the store, A the product, $\mu_R$ reflects to what extent the number of pieces of the product ordered is moderate, and $\mu_S$ to what extent the price is medium or the product is important, depending on the interpretation we are interested in. Note that $\mu_R$ and $\mu_S$ are induced by the query and are not part of the data here.

According to the second choice (the degrees in S express importances) which is possible since S is normalized, the result of the division is:

$$\mu_{R[A \div A]S}(x1) = \min(\mu_S(a1) \rightarrow_D \mu_R(x1,a1), \mu_S(a2) \rightarrow_D \mu_R(x1,a2)) = \min(.8, .7) = .7$$
$$\mu_{R[A \div A]S}(x2) = \min(\mu_S(a1) \rightarrow_D \mu_R(x2,a1), \mu_S(a2) \rightarrow_D \mu_R(x2,a2)) = \min(.5, .6) = .5$$

x2 is weakly selected although it is not related to a2 in R, because a2 is only marginally important in S. On the contrary, with the first interpretation, we get:

$$\mu_{R[A \div A]S}(x1) = \min(\mu_S(a1) \rightarrow_G \mu_R(x1,a1), \mu_S(a2) \rightarrow_G \mu_R(x1,a2)) = \min(.8, 1) = .8$$
$$\mu_{R[A \div A]S}(x2) = \min(\mu_S(a1) \rightarrow_G \mu_R(x2,a1), \mu_S(a2) \rightarrow_G \mu_R(x2,a2)) = \min(.5, 0) = 0.$$

As announced the Gödel implication-based interpretation completely rejects subtuples which are not at all associated with some subtuple s in S (e.g., x2) in the above example. This is not the case with the other interpretation (based on Dienes implication) when the level of importance $\mu_S(s)$ is strictly smaller than 1.

## 4. Division of Fuzzy Relations and Relational Query Languages

### *4.1. Algebraic Aspects*

The two previous interpretations are clearly generalizations of expressions (1)-(2), but also of formula (3), if we choose for the containment operation an inclusion index based on an implication ($\rightarrow$), i.e.:

$$\mathrm{Inc}(A \subseteq B) = \min_{x \in X} \mu_A(x) \rightarrow \mu_B(x). \tag{13}$$

At that point, if we assume that the algebraic operations are defined for fuzzy relations, the status of expression (5) with respect to its ability to express an extended division must be examined. First, let us note that in this expression, the term R[X] is used to define a non-fuzzy referential of X-values. Consequently, if we consider this expression when the relations R and S are possibly fuzzy, we cannot use an extended projection:

$$\mu_{R[X]}(u) = \max_{r \in R \wedge r[X]=u} \mu_R(r)$$

since its result is fuzzy. We need a specific operator, denoted R]X[ returning the support of the projection of the relation R on the set of attributes X. Let us define the support, set difference and Cartesian product for fuzzy relations as follows:

– $\mu_{R]X[}(x) = 1$ if $\exists x$ such that $x = r[X]$ and $\mu_R(r) > 0$, $\mu_{R]X[}(x) = 0$ otherwise,
– $\mu_{R-S}(u) = \min(\mu_R(u), 1 - \mu_S(u))$,
– $\mu_{R \times S}(r,s) = \min(\mu_R(r), \mu_S(s))$.

**Proposition 1**. According to the definitions given above, if R (resp. S) is a fuzzy (resp. fuzzy normalized) relation, the equality: $R[A \div A]S = R]X[ - (R]X[ \times S[A] - R)[X]$ holds if the division is based on Dienes implication, i.e.: $\mu_{R[A \div A]S}(x) = \min_s \mu_S(s) \rightarrow_D \mu_R(x,s[A]) = \min_s \max(1 - \mu_S(s), \mu_R(x,s[A]))$.

**Proof**. Let us denote $R_x$ the tuples of R whose X-value is x (each tuple has the form $(x, a_i) / \mu_i$), similarly, let us write $(y_i, a_i) / \mu'_i$ an S-tuple, where $\mu_i$ and $\mu'_i$ are the attached grades. Then, the evaluation of the degree of x according to formula (5) revisited ($\mu_{R]X[ - (R]X[ \times S[A] - R)[X]}(x)$) requires to calculate:

– $R_x]X[ \times S[A] = \cup_i ((x, a_i) / \mu'_i)$,

– $R_x]X[ \times S[A] - R_x = \cup_i ((x, a_i) / \min(\mu'_i, 1 - \mu_i))$,
– $(R_x]X[ \times S[A] - R_x)[X] = x / \max_i \min(\mu'_i, 1 - \mu_i)$,
– $R_x]X[ - (R_x]X[ \times S[A] - R_x)[X] = x / [1 - \max_i \min(\mu'_i, 1 - \mu_i)] = x / \min_i \max(1 - \mu'_i, \mu_i)$.

This value is exactly that returned by formula (12)♦

This result generalizes formula (5) that is valid for regular relations.

We will now point out some properties of the interpretations of the extended division depending on the chosen implication. The result of the usual division of R(A,X) by S(A,Y) is a set of X-values included in those appearing in R. This result is quite obvious, since it stems from expression (5). Let us examine the situation in the fuzzy case.

**Proposition 2**. If R and S are fuzzy relations, the inclusion $R[A \div A]S \subseteq R[X]$ (i.e., $\mu_{R[A \div A]S}(x) \leq \mu_{R[X]}(x)\ \forall x$) holds for both Dienes and Gödel implications, provided that S is normalized. If S is not normalized the inclusion does not hold.

**Proof**.

***a) Dienes implication***

Consider the definition of the division given in (12). Due to the normalization of S, there exists a tuple $s_0$ in S such that $\mu_S(s_0) = 1$ and

$$\begin{aligned}\mu_{R[A \div A]S}(x) &= \min_s \max(1 - \mu_S(s), \mu_R(x, s[A])) \\ &\leq \max(1 - \mu_S(s_0), \mu_R(x, s_0[A])) \\ &= \mu_R(x, s_0[A]) \leq \max_{r \in R \wedge r[X]=x} \mu_R(r) = \mu_{R[X]}(x).\end{aligned}$$

When S is not normalized counterexamples can be easily exhibited.

*Counterexample*. Let us consider the division defined in (12) and take the following extensions: R = (x, a1) / .2 and S = a1 / .1. In that case, $R[A \div A]S$ = x / .9 which is not included in R[X] (x / .2).

***b) Gödel implication***

When S is normalized, we have, using (11), for s[A] such that $\mu_S(s) = 1$

$$\mu_{R[A \div A]S}(x) \leq \mu_R(x, s[A]) \leq \max_s \mu_R(x, s[A]) = \mu_{R[X]}(x)\ \forall x.$$

The same counterexample applies to Gödel implication, since then $R[A \div A]S$ = x / 1.♦

An illustration of this result in particular can be found in the example given in III.5.

Another interesting property valid for the regular division is the commutativity of the division R[A ÷ A]S and a selection bearing on X by means of a condition denoted by $\varphi_X$ (when S is not empty). In other words, when R and S are regular relations, we have:

$$R[A \div A]S{:}\ \varphi_X = (R{:}\ \varphi_X)[A \div A]S.$$

The question is to know whether this remains true for fuzzy relations and we will examine it for each of the two interpretations presented before.

**Proposition 3**. If R (resp. S) is a fuzzy relation and the division R[A ÷ A]S is based on Dienes or Gödel implication, the equality: $R[A \div A]S{:}\ \varphi_X = (R{:}\ \varphi_X)[A \div A]S$ holds, provided that S is normalized.

**Proof**.

***a) Dienes implication***

Let $a = \mu_{R[A \div A]S:\ \varphi_X}(x) = \min(\mu_{\varphi_X}(x), \min_s \max(1 - \mu_S(s), \mu_R(x, s[A])))$.

Let $b = \min_s \max(1 - \mu_S(s), \min(\mu_R(x, s[A]), \mu_{\varphi_X}(x)))$

Let $\alpha = \mu_{\varphi_X}(x)$. Note that a = b holds when $\alpha = 1$ and also when $\alpha = 0$ since $\exists s, \mu_S(s) = 1$. In the following, assume $0 < \alpha < 1$.

Then $a = \min_s \max(\min(\alpha, 1 - \mu_S(s)), \min(\alpha, \mu_R(x, s[A])))$

$\leq \min_s \max(1 - \mu_S(s), \min(\alpha, \mu_R(x, s[A]))) = b.$

Noticing that $\exists s, \mu_S(s) = 1$, the set $\{s, \mu_S(s) > 1 - \alpha\} \neq \emptyset$, and

$$\begin{aligned} a &= \min(\min_{\mu_S(s) \leq 1-\alpha} \max(\alpha, \min(\alpha, \mu_R(x, s[A]))), \\ &\qquad \min_{\mu_S(s) > 1-\alpha} \max(1 - \mu_S(s), \min(\alpha, \mu_R(x, s[A])))) \\ &= \min_{\mu_S(s) > 1-\alpha} \max(1 - \mu_S(s), \min(\alpha, \mu_R(x, s[A]))) \leq \alpha \\ &\qquad (\text{since } \max(\alpha, \min(\alpha, \mu_R(x, s[A]))) = \alpha). \end{aligned}$$

But $\min_{\mu_S(s) > 1-\alpha} \max(1 - \mu_S(s), \min(\alpha, \mu_R(x, s[A]))) \geq b$.

Then $a \geq b$ and finally $a = b$.

***b) Gödel implication***

Let $\alpha = \varphi_X(x)$. If $\alpha = 1$ the result trivially holds. Assume $\alpha < 1$. We have to prove that

$$\min(\alpha, \min_s \mu_S(s) \to_G \mu_R(x, s[A]) = \min_s \mu_S(s) \to_G \min(\alpha, \mu_R(x, s[A]).$$

Since S is normalized, the second expression reads

$$\min_{s:\mu_S(s)>\min(\alpha,\mu_R(x,s[A])}\min(\alpha, \mu_R(x, s[A]))$$
$$= \min(\alpha, \min_{s:\mu_S(s)>\min(\alpha,\mu_R(x,s[A])}\mu_R(x, s[A]))$$
$$= \min(\alpha, \min_{s:\mu_S(s)>\alpha}\mu_R(x, s[A]), \min_{s:\mu_S(s)>\mu_R(x,s[A])}\mu_R(x, s[A]))$$
$$= \min(\alpha, \min_{s:\mu_S(s)>\alpha}\mu_R(x, s[A]), \min_s \mu_S(s) \rightarrow_G \mu_R(x, s[A])).$$

Let $a = \min_s \mu_S(s) \rightarrow_G \mu_R(x, s[A])$, $b = \min_{s:\mu_S(s)>\alpha}\mu_R(x, s[A])$.

We have to prove $\min(\alpha,a) = \min(a, b, \alpha)$, i.e., $b \geq \min(a,\alpha)$.

Assume $\{s, \mu_S(s) > \mu_R(x, s[A])\} = \emptyset$ then $a = 1$, and $b \geq \alpha$ (since then $\mu_R(x, s[A]) \geq \mu_S(s)$, $\forall s$) and the equality holds. Now assume this set is not empty, and that $b < \alpha$ (the case $b \geq \alpha$ is then trivial). This means that if $\mu_R(x, s[A])$ is minimum in s* on $\{s, \mu_S(s) > \alpha\}$, it holds that $\mu_S(s^*) > \alpha > \mu_R(x, s^*[A])$ (= b).

Hence

$$a = \min_{s:\mu_S(s)>\mu_R(x,s[A])}\mu_R(x, s[A])$$
$$\leq \min_{s:\mu_S(s)>\alpha>\mu_R(x,s[A])}\mu_R(x, s[A]) = \mu_R(x, s^*[A]) = b. \blacklozenge$$

When the assumption that S is normalized is not made, Proposition 3 generally fails when $\varphi_X(x) < 1$. Using Dienes implication, a counterexample is obtained if the set $\{s, \mu_S(s) > 1 - \alpha\}$ is empty. Especially if $\mu_S(s) < 1 - \alpha$, $\forall s$; then $\max(1 - \mu_S(s), \min(\alpha, \mu_R(x, s[A]))) > \alpha = \min(\alpha, \max(1 - \mu_S(s), \mu_R(x, s[A]))$. Using Gödel implication, the counterexample occurs if $\forall s$, $\mu_S(s) \leq \min(\mu_R(x, s[A]), \alpha)$ for $\alpha < 1$ then $\min(\alpha, \mu_S(s) \rightarrow \mu_R(x, s[A])) = \alpha < \mu_S(s) \rightarrow \min(\alpha, \mu_R(x, s[A]) = 1$.

The failure of Proposition 3 when S is not normalized is due to the fact that we are dividing by a set S which is somewhat empty; in such a case, as already said, the user should be notified, when dealing with levels of importance. In case of degrees of fulfillment the user may be also informed; however the division still makes sense in this case, but the system should be careful, since then, some transformations are no longer allowed.

### *4.2. Expressing the Division in SQLf*

SQLf [3], [5] is a relational language supporting fuzzy queries addressed to a regular relational database. It has been designed as an extension of the standard SQL by allowing fuzzy conditions (which define preferences over more or less acceptable values) both at the level of tuple and set of tuples selection. In fact, all the constructs existing in SQL have been kept and new ones have been added in SQLf, in particular to calibrate the size of the answer.

We have seen in Section 2 how queries calling on a division could be written in the SQL language and one question is to know what to do for fuzzy queries. Let us consider expression (6) recalled hereafter:

*select distinct* X *from* R R1 *where not exists*
(*select* * *from* S *where* A *not in* (*select* A *from* R R2 *where* R2.X = R1.X))

and let us assume that R and S are fuzzy relations, one may wonder about the meaning of this expression as far as the operators "not exists" and "not in" are defined for nested blocks delivering fuzzy relations. Such extended definitions for set membership ("in") and set non-emptiness ("exists") nesting operators (and their negations) have been proposed in [5]. Thus, if R is a fuzzy relation, these predicates are defined as follows:

– a) fuzzy set membership: *in* (*select* b *from* R):

$$\mu_{in}(a,R) = \max_{b \in \text{support}(R) \wedge b=a} \mu_R(b). \tag{14}$$

– b) set non emptiness: *exists* (*select* * *from* R):

$$\mu_{exists}(R) = \max_{r \in \text{support}(R)} \mu_R(r). \tag{15}$$

**Proposition 4**. If R (resp. S) is a fuzzy (resp. fuzzy normalized) relation, according to the above expressions for the predicates "in" and "exists", the SQLf expression:

*select distinct* X *from* R R1 *where not exists*
(*select* * *from* S *where* A *not in* (*select* A *from* R R2 *where* R2.X = R1.X))

is equivalent to the division based on Dienes implication.

**Proof**. An X-value x of R receives the degree:

$$\mu(x) = \max_{r \in R \wedge r[X]=x} \min(\mu_R(r), \mu_{not\ exists}(S'))$$

where S' is the result of the nested block involving relation S, i.e., (*select* * *from* S *where* A *not in* (*select* A *from* R R2 *where* R2.X = R1.X)). According to formula (15) we have:

$$\mu_{not\ exists}(S') = 1 - \max_{s \in \text{support}(S')} \mu_{S'}(s) = 1 - \max_{s \in S} \min(\mu_S(s), \mu_{not\ in}(s[A], R'))$$

where R' denotes the result of the innermost nested block. According to formula (14) we have:

$$\begin{aligned}\mu_{\text{not in}}(s[A], R') &= 1 - \max_{a \in \text{support}(R') \wedge a=s[A]} \mu_{R'}(a)\\ &= 1 - \max_{r' \in R \wedge r'[X]=x \wedge r'[A]=s[A]} \mu_R(r')\\ &= 1 - \mu_R(x, s[A]).\end{aligned}$$

Thus:

$$\begin{aligned}\mu(x) &= \max_{r \in R \wedge r[X]=x} \min(\mu_R(r), 1 - \max_{s \in S} \min(\mu_S(s), 1 - \mu_R(x, s[A])))\\ &= \max_{r \in R \wedge r[X]=x} \min(\mu_R(r), \min_{s \in S} \max(1 - \mu_S(s), \mu_R(x, s[A]))).\end{aligned}$$

The supremum is obtained for the tuple $r_0$ of R whose X-value equals x and whose grade in R is maximum, since the other component of the min operation depends only on x. Consequently, the previous expression reduces to:

$$\mu(x) = \min(\mu_R(r_0), \min_{s \in S} \max(1\text{-}\mu_S(s), \mu_R(r_0[X], s[A]))).$$

Since S is normalized, it is easy to show that: $\min_{s \in S} \max(1 - \mu_S(s), \mu_R(r_0[X], s[A])) \leq \mu_R(r_0)$ and:

$\mu(x) = \min_{s \in S} \max(1 - \mu_S(s), \mu_R(r_0[X], s[A]))$ which is what we have in expression (12).

N.B.: If we used the non-commutative conjunction [13] a & b = b if a + b > 1 and a & b = 0 otherwise, instead of min(a,b) in the definition of S', we would finally get $\mu(x) = \max_{r \in R \wedge r[X]=x} \min(\mu_R(r), \min_{s \in S} \mu_S(s) \rightarrow_G \mu_R(x, s[A]))$ where $\rightarrow_G$ is Gödel implication. However, one problem is in the fact that this choice is somewhat hidden (in the implicit conjunction tied to a selection applied to a fuzzy relation) and does not seem to be available to the user. The same remark can be made regarding the algebraic counterpart of this SQLf statement, in the spirit of Proposition 1 for Dienes implication, namely, R]X[ – (R]X[ × S[A] – R) [X], which leads to the interpretation based on Gödel implication if the conjunction involved in the second difference is replaced by the non commutative one given before.♦

Let us now illustrate the use of such a formulation in SQLf in the context of a regular database composed of the relations ORDERS and PRODUCTS presented before. The query: "find the stores which have ordered a moderate amount of pieces for all medium priced products" will be basically written:

*select distinct* store *from* O O1 *where not exists*

(*select * from* P *where* p# *not in*
(*select* p# *from* O O2 *where* O2.store = O1.store)) (Q)

where O represents the orders with a moderate quantity and P the products whose price is medium. These relations could be built as intermediate relations with the help of:

*select ** into O *from* ORDERS *where* qty = 'moderate' (q1)
*select ** into P *from* PRODUCT S *where* price = 'medium'. (q2)

In fact, the sequence of queries q1-q2-Q (which makes fuzzy relations more explicit) can be replaced by:

*select distinct* store *from* ORDERS O1 *where* qty = 'moderate' *and not exists*
(*select * from* PRODUCTS *where* price = 'medium' *and* p# *not in*
(*select* p# *from* ORDERS O2 *where* qty = 'moderate' *and* O2.store = O1.store)). (Q')

Consequently, starting with a regular database involving relations R and S, the expression:

*select distinct* X *from* R R1 *where* fc1 *and not exists*
(*select * from* S *where* fc2 *and* A *not in*
(*select* A *from* R R2 *where* fc1 *and* R2.X = R1.X)) (16)

where fc1 (resp. fc2) is a fuzzy condition applying to R (resp. S), results in the division of R by S according to Dienes implication as far as at least one S-tuple satisfies completely fc2 i.e., the fuzzy relation involved in the division is normalized.

A last question deals with the ability to recover the interpretation of the extended division based on Gödel implication. It is easy to see that if we use the approach proposed in formula (13) to interpret expression (7), it will be possible to reach in particular the two meanings considered here which are based on Dienes and Gödel implication. The corresponding expression (analogous to (16)) in SQLf is then:

*select* X *from* R *where* fc1
*group by* X
*having set*(A) *contains*$_{D|G}$ (*select* A *from* S *where* fc2)

where contains$_{D|G}$ stands for a containment interpreted in the spirit of (13):

$$\text{R contains}_{D|G}\ \text{S} = \min_{x \in X} \mu_S(x) \rightarrow_{D|G} \mu_R(x)$$

depending on the intended meaning of the division. It should be remarked that this syntactic framework is specific to the division of fuzzy relations (thanks to the use of the operator $\text{contains}_D$) unlike the one given in expression (16) (it turns out that some queries very similar to (16) do not call on a division). For this reason, the use of the construct based on "$\text{contains}_D$" allows to check if its right-hand side relation is normalized. If not, the user will be informed of the fact and asked to choose either to withdraw his query, or to modify it, or to force the normalization of the concerned relation according to a transformation such as: $\mu'_S(s) = \mu_S(s) / \max_S \mu_S(s)$ which preserves the notion of importance attached to the grades (an alternate solution can be to change $\max_S \mu(s)$ into 1). Since the normalization cannot be guaranteed with expression (16), we suggest to discard this type of expression for a division. Finally, using the "contains" operator, the previous query: "find the stores which have ordered a moderate amount of pieces for all medium priced products" can be written:

*select* store *from* ORDERS *where* qty = 'moderate' *group by* store
*having set*(p#) *contains*$_{D|G}$ (*select* p# *from* PRODUCTS *where* price = 'medium').

## 5. Conclusion

This paper is mainly concerned with the extension of the division operation, in the context of relational database management systems, to fuzzy relations (for instance, intermediate relations obtained from base relations when processing fuzzy queries), where a grade is tied to each tuple. A point (of prime importance for the user) concerns the possible meanings of the extended operation. Two basic interpretations have been identified which depend on the semantics of the grades. In the division of a relation R by a relation S, R-grades are assumed to express degrees of fulfillment of a condition whereas S-grades may either express degrees of fulfillment, or levels of importance. Formal definitions of the two types of division, based on Dienes implication and Gödel implication respectively, have been proposed and the validity of some properties which hold in the usual case has been examined. It turns out that they are still valid for the two extensions, provided that the set of S-grades is normalized. When S is not normalized, we are dividing by a set that is somewhat empty, which clearly makes problems.

Finally, the expression of divisions in the framework of SQLf, a language supporting fuzzy querying, has also been investigated. We have shown that one classical expression of the division in terms of "in" and "exists" nestings in SQL led to the algebraic interpretation based on Dienes implication provided that S is normalized. The other interpretation (relying on Gödel implication) can be attained only with a special construct of SQLf (set containment along with a grouping), insofar as only min-conjunction is used.

The evaluation of queries involving an extended division (whatever its interpretation) has not yet been addressed. It is worth noticing that the evaluation of the classical division has not been investigated in the literature due to the non-primitivity of the operation. For the extended division a first algorithmic approach would consist in a sorting of the R relation in order to build a partition (according to X-values) and then to search in each subset of this partition for the A-values present in S so as to compute the value of the implication. The cost of such a process (in terms of page accesses) is that of a sort (M log M if M is the number of pages of R) plus that of a Cartesian product between S and R (roughly speaking, the cost is $N \times M$ if N is the number of pages of S). However, improvements to this first view must be investigated (e.g., with Gödel implication, a subset can be discarded as soon as it does not contain some A-value of S). The design and implementation of refined algorithms performing the division according to the two basic interpretations proposed in this paper remain a line for future research.

## References

[1] W. Bandler and L.J. Kohout, Fuzzy relational products as a tool for analysis and synthesis of the behavior of complex natural and artificial systems, in: P.P. Wang and S.K. Chang, eds., *Fuzzy Sets — Theory and Applications to Policy Analysis and Information Systems* (Plenum Press, New York, 1980) 341-367.

[2] D. Barbará, H. Garcia-Molina and D. Porter, The management of probabilistic data, *IEEE Trans. on Knowledge and Data Engineering* **4**(5) (1992) 487-502.

[3] P. Bosc, M. Galibourg and G. Hamon, Fuzzy querying with SQL: Extensions and implementation aspects, *Fuzzy Sets and Systems* **28** (1988) 333-349.

[4] P. Bosc and O. Pivert, Some approaches for relational databases flexible querying, *J. of Intelligent Information Systems* **1** (1992) 323-354.

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

[6] B.P. Buckles and F.E. Petry, A fuzzy representation of data for relational databases, *Fuzzy Sets and Systems* **5** (1982) 213-226.

[7] D. Chamberlin et al., SEQUEL2: A unified approach to data definition, manipulation and control, *IBM J. of Research and Development* **20** (1976) 560-575.

[8] E.F. Codd, A relational model of data for large shared data banks, *Communication of the ACM* **13** (1970) 377-387.

[9] B. De Baets and E.E. Kerre, Fuzzy relational compositions, *Fuzzy Sets and Systems* **60** (1993) 109-120.

[10] D. Dubois and H. Prade, Weighted minimum and maximum operations in fuzzy set theory, *Information Sciences* **39** (1986) 205-210.

[11] D. Dubois and H. Prade (with the collaboration of H. Farreny, R. Martin-Clouaire and C. Testemale), *Possibility Theory — An Approach to Computerized Processing of Uncertainty* (Plenum Press, New York, 1988).

[12] D. Dubois and H. Prade, Measuring properties of fuzzy sets: A general technique and its use in fuzzy query evaluation, *Fuzzy Sets and Systems* **38** (1990) 137-152.

[13] D. Dubois and H. Prade, Fuzzy sets in approximate reasoning — Part 1: Inference with possibility distributions, *Fuzzy Sets and Systems* **40** (1991) 143-202.
[14] D. Dubois and H. Prade, Upper and lower images of a fuzzy set induced by a fuzzy relation: Application to fuzzy inference and diagnosis, *Information Sciences* **64** (1992) 203-232.
[15] D. Dubois and H. Prade, Quotient operators in fuzzy relational databases, in: *Proc. of the 2nd Europ. Congress on Fuzzy and Intelligent Techniques (EUFIT'94)*, Aachen, Germany, 1994a, 357-360. Revised version entitled: Semantics of quotient operators in fuzzy relational databases, to appear in: *Fuzzy Sets and Systems*.
[16] D. Dubois, H. Prade and P. Smets Partial truth is not uncertainty: Fuzzy logic versus possibilistic logic, *IEEE Expert* **9** (1994b) 15-19.
[17] W. Kim, On optimizing an SQL-like nested queries, *ACM Trans. on Database Systems* **7** (1986) 443-469.
[18] N. Mouaddib, The nuanced relational division, in: *Proc. of the 2nd IEEE Inter. Conf. on Fuzzy Systems (FUZZ-IEEE'93)*, San Francisco, CA, 1993, 1419-1424.
[19] H. Prade and C. Testemale, Generalizing database relational algebra for the treatment of incomplete/uncertain information and vague queries, *Information Sciences* **34** (1984) 115-143.
[20] M. Umano, FREEDOM-0: a fuzzy database system, in: M.M. Gupta and E. Sanchez eds., *Fuzzy Information and Decision Processes* (North-Holland, Amsterdam, 1982) 339-347.
[21] M. Umano and S. Fukami, Fuzzy relational algebra for possibility-distribution-fuzzy-relational model of fuzzy data, *J. of Intelligent Information Systems* **3** (1994) 7-28.
[22] R. Vandenberghe, A. Van Schooten, R. De Caluwe and E.E. Kerre, Some practical aspects of fuzzy database techniques: An example, *Information Systems* **14** (1989) 465-472.
[23] E.A. Wong, Statistical approach to incomplete information in database systems, *ACM Trans. on Database Systems* **7** (1982) 470-488.
[24] R.R. Yager, On ordered weighted averaging aggregation operators in multicriteria decisionmaking, *IEEE Trans. on Systems, Man and Cybernetics* **18** (1988) 183-190.
[25] R.R. Yager, Fuzzy quotient operators for fuzzy relational databases, in: *Proc. of the Inter. Fuzzy Engineering Symp. (IFES'91)*, Yokohama, Japan, 1991, 289-296.
[26] L.A. Zadeh, Fuzzy sets, *Information and Control* **8** (1965) 338-353.