Extending the Relational Model with Uncertainty and Ignorance

Sunil Choenni, Henk Ernst Blok, and Maarten Fokkinga

University of Twente
Faculty EEMCS
P.O.Box 217, 7500AE Enschede
The Netherlands
{r.s.choenni, h.e.blok, m.m.fokkinga}@utwente.nl

Abstract.
It has been widely recognized that in many real-life database applications there is growing demand to model uncertainty and ignorance. However the relational model does not provide this possibility. Through the years a number of efforts has been devoted to the capture of uncertainty and ignorance in databases. Most of these efforts attempted to capture uncertainty using the classic probability theory. As a consequence, the limitations of probability theory are inherited by these approaches, such as the problem of information loss.
In this paper, we extend the relational model with uncertainty and ignorance without these limitations posed by the other approaches. Our approach is based on the so-called theory of belief functions, which may be considered as a generalization of probability theory. Belief functions have an attractive mathematical underpinning and many intuitively appealing properties.

1. Introduction

Uncertainty is inherent in many real-life systems. In order to support these systems a comprehensive handling of uncertain data is required. However, the relational model ignores all uncertain data, and therefore, does not deliver satisfactory results in many cases. For example, in a military setting, a database may be used to monitor the locations of battleships. These locations typically are the product of intelligence gathering. In a business setting, marketing and production decisions are mostly based on expected customer behaviour which is never deterministic. In a medical setting, doctors would like to store the most plausible diseases for a patient. Many similar examples can be found in the literature, all pointing towards the extension of the relational model with uncertainty [BaGP 90, BhKa 86, DeSa 96, GuKT 91, Wong 82].
Through the years a number of efforts has been devoted to the capture of uncertainty and ignorance in databases. In general uncertainty can be modelled by logical and statistical approaches in adequate way, while this is not the case for ignorance. The approaches classified as logical use logic and inference rules to handle uncertainty [GuKT 91, LaSA 94, RaMa 88]. The earliest form of logic was the use of three-valued logic, i.e., true, false, and unknown. Multi-valued logic is also used to handle uncertainty. It is clear that even with multi-valued logic the number of combinations for events is still finite. To deal with this limitation, fuzzy set theory has been exploited to handle uncertainty [RaMa 88].
The approaches classified as statistical often express uncertainty in the form of a probability [BaGP 90, CaPi 87, Fuhr 93, GeHe 86, Lee 92, DeSa 96]. These approaches in turn can be divided in two categories. In the first category, uncertainty is handled at a tuple level, i.e., a probability value is assigned to a tuple of a relation [CaPi 87, GeHe 86, Lee 92, DeSa 96]. In the second category, uncertainty is handled at attribute level [BaGP 90]. This paper is devoted to uncertainty handling at attribute level.
In the statistical approaches, except in [Lee 92], probability theory is used to extend the relational model. As a consequence the limitations of this theory are inherited by these extended models. One of the major drawbacks of probability theory is that it does not leave room to model ignorance, while this notion is inherent to many real-life systems. This limitation has been noted and discussed in [BaGP 90, Lee 92]. To overcome these limitations, Barbara et al. [BaGP 90] introduced the so-called notion of missing probability, which is actually a way to model ignorance. Suppose we want to reason about the weather and we consider **Ω** = {dry, wet} as the set of possible outcomes. Now, we have collected evidences that it will be a dry season with probability 0.6 and another set of evidences is pointing to a wet season with a probability 0.2. Since the probability of dry and wet season sum up to 0.8 (0.6 + 0.2), the remaining 0.2 is called missing probability. Although the paper [BaGP 90] claims that the introduced extended relational model is based on probability theory, it should be clear that this is not the case due to the introduction of missing probability. Recall that the sum of the probabilities of all possible outcomes should be 1.0, i.e., the sum of the probability of dry and wet season should be 1.0 in our example according to probability theory. Since this is not the case, one of the axioms of probability theory is violated (see Section 2.1). Furthermore, the authors of [BaGP 90] observed that their approach leads to the problem of

information loss, which is a consequence of an inadequate way of joining two relations on an attribute to which a kind of probability function is associated. Let us extend our weather example in order to illustrate the concept of information loss. Assume that we have a database that keeps track of the crop planting behaviour of farmers. In a dry season, 30% of the farmers choose to plant turnips and 70% choose to plant wheat and in a wet season all farmers decide to plant turnips. The crop planting behaviour data and the weather data are modelled in Figure 1.1.

source	weather
KNMI	0.6 [dry] 0.2 [wet] 0.2 [*]

weather	plant
dry	0.3 [turnips] 0.7 [wheat]
wet	1.0 [turnips]

Figure 1.1: Two base relations to model the weather data

We note that KNMI is an abbreviation of the Royal Netherlands Meteorological Institute.
Joining these relations on attribute weather according to a probabilistic approach results in the table shown in Figure 1.2.

source	weather plant
KNMI	0.18 [dry turnips] 0.42 [dry wheat]
KNMI	0.2 [wet turnips]

Figure 1.2: Result of a join between the base relations depicted in Figure 1.1

As can be verified from the resulting table, the missing probability of 0.2 has no influence on the join result. So we have this information in one of our base table but it is not used during the join, hence we have information loss.
In [Lee 92], Dempster-Shafer theory [Demp 67, Shaf 76] is applied to handle uncertainty in relational databases.
In artificial Intelligence, this theory has attracted much attention and variants of the theory have been applied to handle uncertain information in many applications, ranging from expert systems to sensor data fusion. In [Lee 92] this theory is exploited to answer questions like "What is the belief that two persons have the same blood type?", assuming that the blood type is recorded as a stochastic attribute in a database or "What is the belief that two tuples with almost the same values in a database represent the same person?".
In this paper, we extend the relational model with uncertainty and ignorance at attribute level by means of Dempster- Shafer theory, also known as the theory of belief functions. The motivation to use the theory of belief functions to model uncertainty and ignorance is that it provides an intuitive and attractive way to model these issues. For example, it is easy to model uncertain facts like "The belief in A on the basis of evidence E". A central instrument in this theory is the combination rule of Dempster that is used to combine belief functions. Furthermore, belief functions have an attractive mathematical underpinning and many intuitively appealing properties.
Actually in this paper we build on on the pioneering work of Barbara et al. [BaGP 90]. We provide a theoretical basis for the approach that has been taken by Barbara et al. and solve the problem of information loss. The theoretical underpinning of the approach is based on the theory of belief functions, which may be considered as a generalization of probability theory [Demp 68, HaFa 92]. We adapt the combination rule of Dempster in order to perform joins in an adequate way.
The remainder of this paper is organized as follows. In Section 2 we discuss our extended relational model after a brief introduction of belief functions. Then in Section 3 we discuss the impact of our extension on the basic relational operators; the PROJECTION, SELECTION, and JOIN operator. Finally, in Section 4 we conclude the paper.

2. Extended Relational Model

This section is devoted to an extension of the relational model. In our extended model, an attribute in a tuple may assume a set of values. As a consequence, such a set contains a subset of the domain of an attribute. We represent ignorance ---with regard to the value of an attribute A in a tuple--- by assigning the (whole) domain of A as value for A in the tuple. Furthermore, to each attribute value in a tuple an uncertainty measure is attached. In Section 2.1, we briefly discuss the formal semantics of uncertainty and ignorance as used in the theory of belief functions. Then, in Section 2.2, we extend the relational model with these notions. Finally, in Section 2.3, we define the class of relational schemes to be considered for discussion of our solution for the problem of information loss.

2.1 Introduction to belief functions

A belief function is a function that is defined on a frame of discernment, which is a set of mutually exclusive and exhaustive hypotheses about some problem domain. Before introducing the definition of a belief function, we introduce the definition of basic probability assignment.

[**Def. 2.1**] Let set D be a frame of discernment, then a function $m : 2^D \rightarrow [0,1]$ is called a basic probability assignment, abbreviated as bpa, whenever $m(\emptyset) = 0$ and $\sum_{S \subseteq D} m(S) = 1$.

The quantity $m(S)$ expresses a relative confidence in exactly S and not in any (proper) subset of *S*. The total confidence in *S*, which we call *belief*, is the sum of the probability assignments committed to all subsets of *S*. The following definition describes the relation between a belief function and basic probability assignment.

[**Def. 2.2**] For a given bpa *m*, a belief function, called *Bel*, is defined over any set $S \subseteq D$ as $Bel(S) = \sum_{S' \subseteq S} m(S')$.

We note that a bpa induces a belief function and conversely. In order to define the notion of ignorance, we first define plausibility.

[**Def. 2.3**] The plausibility of any set $S \subseteq D$ is defined as $Pl(S) = 1 - Bel(\neg S)$.

[**Def. 2.4**] The degree of ignorance for any set $S \subseteq D$ is defined as $Ig(S) = Pl(S) - Bel(S)$.

Let us illustrate the above-mentioned definitions by means of an example in the context of relational databases.

[**Example 2.1**] Suppose that a department store wants to record the expected performance of its employees working in different departments in a relation called EMPLOYEE. Let us assume that the department in which an employee works, is deterministic and that the expected sale by an employee is an uncertain attribute, i.e., the sale attribute assumes a set of values, and a bpa, *m*, is associated to each set of values. This might be the case if the set of values for the sales attribute is determined on the basis of some bodies of evidence. Furthermore, attribute Name is the key of the relation. An instance of such a relation is given below.

Name	Department	Sales
Jon Smith	Toy	0.3 [€ 30-34K] 0.2 [€ 32-33K] 0.3 [€ 35-39K] 0.2. [*]
Joe Sixpack	Clothes	0.9 [€ 30-34K] 0.1[*]

The meaning of the first tuple is that Jon Smith is working in the Toy department and we expect with a bpa of 0.3 that his sale will be between 30 and 34K euro's , with a bpa of 0.2 that his sale will be between 32 and 33K, and with a bpa of 0.3 that his sale will be between 35 and 39K euro's. Let us assume that the domain of Sales is *DS* = {30K, 31K,… ,39K}. Then, we assign a bpa of 0.2 to the whole set DS, expressed by [*], since we do not have any clue for the sales value. This is a way to model ignorance, i.e., there are no evidences to prefer a specific element or range of DS above another one.
From the bpa's in relation EMPLOYEE we can derive the following belief and plausibility values and degrees of ignorance with regard to the sales of Jon Smith.

Sales range	*Bel*	*Pl*	*Ig*
[€ 30-34K]	0.5	0.7	0.2
[€ 32-33K]	0.2	0.7	0.5
[€ 35-39K]	0.3	0.5	0.2
[*]	1.0	1.0	0

We note that *Bel*([€ 30-34K])[1] = *m*([€ 30-34K]) + *m*([€ 32-33K]) = 0.3 +0.2 = 0.5. Since the range [€ 35-39K] is the only range that is definitely disjoint with the range [€ 30-34K], the *Pl* value for [€ 30-34K] is 1- 0.3 = 0.7. Consequently, *Ig*([€ 30-34K]) = 0.7 -0.5 = 0.2. The belief, plausibility and ignorance values for the remaining ranges may be computed in the same way.
[*End of Example 2.1*]

We note that probability theory does not allow us to model ignorance as has been done in relation EMPLOYEE. We can easily prove that the distribution function in the Sales column does not meet the basic axioms of probability theory. Recall that a probability measure is defined on a set (Ω, Γ), in which Ω is the set of possible outcomes and Γ contains the events in whose occurrences we may be interested. The probability function P is defined as $P : \Gamma \to [0,1]$. Let us assume that in the above-mentioned example the set of possible outcomes for Sales is $\Omega = \{30K, 31K, 32K, ..., 39K\}$ and Γ consists of the following set of ranges Γ = {[€ 30-34K], [€ 32-33K], [€ 35-39K],[*]}. Then, P([€ 30-34K] ∪ [€ 35-39K]) = P(Ω). Since P([€ 30-34K] ∪ [€ 35-39K]) = 0.6 and not 1, this violates the axiom P(Ω) = 1 as is required by probability theory. Therefore, probability theory can not be used to model ignorance as defined above. Furthermore, if we still do attempt to capture uncertainty and ignorance by means of probability theory, despite of one basic axiom not being met, this leads to problems, e.g., information loss as illustrated in [BaGP 92].

2.2 Capturing Uncertainty and Ignorance in the Relational Model

We apply the theory of belief functions straightforward in order to model uncertainty and ignorance in relational databases. To each attribute that has an uncertain nature a basic probability assignment is associated. Therefore, we

[1] Formally, we have to write *Bel*({[€ 30-34K]}). However, if a set contains one element, we omit the set brackets for convenience's sake.

will revise the definition of a relational scheme. First we define what is understood by a partition and then we introduce a revised definition of a relational scheme.

[**Def. 2.5**] Let A be a set of attributes $\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_n$. A set K ={ $K_1, K_2, K_3, \ldots, K_l$ } ($l \le n$) is a partition of A whenever $\forall i, j : K_i \cap K_j = \varnothing$ and $\bigcup_i K_i = A$. The domain of $K_i = \{\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_j\}$ is defined as $dom(\alpha_1) \times dom(\alpha_2) \times dom(\alpha_3) \times \ldots \times dom(\alpha_j)$ in which $dom(\alpha_h)$ is the domain of α_h.

[**Def. 2.6**] Let K = { $K_1, K_2, K_3, \ldots, K_l$ } be a partition of a set of attributes of A = $\{\alpha_1, \alpha_2, \alpha_3, \ldots \alpha_n\}$. A relation R is defined over a partition $K_1, K_2, K_3, \ldots, K_l$ and is a subset of the Cartesian product $U_1 \times U_2 \times \ldots \times U_n$, in which U_j is defined as follows

$$U_j = \begin{cases} \{(S, m_j(S)) \mid S \subseteq dom(K_j)\} & \text{if } K_j \text{ is stochastic} \\ \\ dom(K_j) & \text{otherwise} \end{cases}$$

and $dom(K_j)$ is the domain of K_j. A relational scheme is defined as a set of relations.

Consequently we redefine a tuple as a *list of sets of (multi) attribute values* to which a bpa is associated. The set of attribute values of non stochastic attributes consists of one element and the bpa is 1.0. For convenience's sake we omit this bpa function for non stochastic attributes.

We note that if $l = n$ in Def. 2.6, then each K_i consists of exactly one attribute α. This means that bpa's are assigned to single attributes, e.g., as in the relation EMPLOYEE in Example 2.1.

If $l < n$, there is at least one bpa that is defined over a multi-attribute. Consider a set of attributes A = {source, weather, plant} as introduced in the context of our weather example in Section 1. If we choose a partition K = $\{K_1, K_2\}$, in which $K_1 = \{\text{source}\}$ and $K_2 = \{\text{weather, plant}\}$, then K_2 is an example of a multi-attribute. The relation corresponding to Figure 1.2 is defined on three attributes which in turn are divided over two partitions. So, $l < n$ for this case.

The extension of the relational model in the above-mentioned way implies that relations are not longer in first normal form. Furthermore, the extended model gives rise to a new type of questions that are directly related to the bpa attached to an attribute. For example, a question in the context of the example in Section 2.1 might be "Give me the name of the employees for which the belief is greater than 0.3 that their sales will be 35K or higher". This type of queries is a compound selection on a bpa and on the sales attribute. In Section 3.1, we discuss how to deal with this type of queries.

Another interesting type of query might be "What is the belief that Jon Smith and Joe Sixpack will have the same turnover". How to deal with this type of queries is extensively discussed in [Lee 92] and is beyond the scope of this paper.

2.3 Problem limitations

Although we introduced in the previous section a comprehensive definition of a relational scheme, we restrict our selves to a special class of relational schemes in our further discussion. The reason is that our primary goal is discussing of our solution for the problem of information loss.

The class of relational schemes that we consider, consists of a set of *base* relations, in which bpa's are defined on *single* attributes. Furthermore, we assume that there is a non stochastic attribute that serves as key and uniquely identifies a tuple in a *base* relation. All other attributes in the relation, including their bpa's, are dependent on the key. The operators that we consider for the time-being are selection, projection and join and an operator is defined only on *base* relations.

The relations in Figure 1.1 are typical examples of base relations. We note that the relation in Figure 1.2 is not a base relation but a result of a join between the two base relations depicted in Figure 1.1. Hence we restrict ourselves to this type of join results in the remainder of this paper.

3. Relational Operators

This section is devoted to the impact of the extended relational model on the well-known relational operators, PROJECTION, SELECTION, and JOIN. In Section 3.1, we redefine the PROJECTION operator. Section 3.2 is devoted to the redefinition of the SELECTION operator, and illustrates the potential of this operator in the extended relational model. Section 3.3 is devoted the join operator. We introduce an extension on the so-called combination rule of Dempster to combine bpa's in order to support a join.

3.1 Projection

The projection operator in the extended model is quite similar to the projection operator in the relational model. The projection on a stochastic attribute delivers the attribute values together with the bpa associated with the values. For example, the projection ∏(Name, Sales) on the relation EMPLOYEE in Example 2.1 results in

Name	Sales
Jon Smith	0.3 [€ 30-34K] 0.2 [€ 32-33K] 0.3 [€ 35-39K] 0.2. [*]
Joe Sixpack	0.9 [€ 30-34K] 0.1[*]

3.2 Selection

The extension of the relational model with uncertainty gives rise to a new type of selection besides the conventional well-known selection operator. However, the result of a selection remains a (derived) relation. The new type of selection pertains to the bpa values. In the WHERE clause of a query it is allowed to put a restriction on a bpa value. Such a restriction can be posed explicitly on a bpa or on an implicit way. An example of a query that explicitly poses a restriction on a bpa, in the context of the Employee relation introduced in Section 2, is:

SELECT* **FROM** Employee **WHERE** *bpa*(Sales > 35K) ≥ 0.3. The result of this selection is the following relation:

Name	Department	Sales
Jon Smith	Toy	0.3 [€ 30-34K] 0.2 [€ 32-33K] 0.3 [€ 35-39K] 0.2. [*]

Actually the above-mentioned restriction in the **WHERE** clause can be considered as a conjunction of two predicates (Sales >35 K) and (bpa(Sales) ≥ 0.3), respectively. In the relation Employee, only the first tuple meets both restrictions, and hence the result.
We note that, although the following result may look appealing, it is not correct, since the sum of the bpa values is not 1, and therefore Def. 2.1is violated.

Name	Department	Sales
Jon Smith	Toy	0.3 [€ 35-39K]

To allow such a result, operators need to be defined at a nested level and the bpa values should be taken care of. However this beyond the scope of this paper.

Furthermore, we can also pose also restrictions on the belief or plausibility of a sales value, i.e., we implicitly pose restrictions on a bpa. For example, the restriction *Pl*(Sales > 35K) ≥ 0 in the **WHERE** clause will result in the relation:

Name	Department	Sales	*Pl*
Jon Smith	Toy	0.3 [€ 30-34K] 0.2 [€ 32-33K] 0.3 [€ 35-39K] 0.2. [*]	0.7 0.7 0.5 1.0
Joe Sixpack	Clothes	0.9 [€ 30-34K] 0.1[*]	1.0 1.0

For convenience's sake, we attached the plausibility values of the sales ranges as an additional column, *Pl*, to the result. It should be clear that the first and second tuple qualify as result since the *Pl*(Sales > 35K) values for these tuples are 0.5 and 0.1 respectively.
Note that the **WHERE** clause of a query may contain an arbitrary range for the attribute Sales. The belief and plausibility values for such a range can be computed according to Def. 2.2 and Def. 2.3, respectively.
For example, the result of a query:
SELECT * **FROM** Employee **WHERE** *Bel*([30-35K]) > 0.6 results in:

Name	Department	Sales
Joe Sixpack	Clothes	0.9 [€ 30-34K] 0.1[*]

We note that although the bpa of the range [30-35] is 0, the belief value of this range is 0.9.
In some applications it might be sensible to define aggregation and other (nested) operators within a non-stochastic attribute. However, as stated before, the study of this type of selections is a topic for further research.

3.3 Join

This subsection is devoted to the role of the join in our extended relational model. The result of a join can be regarded as a selection on the Cartesian product of a set of relations. Due to page limitations, we restrict ourselves to an equi-join between two base relations. We discuss how to combine bpa's in order to obtain acceptable results for an equi-join. The following example, adopted from [BaGP 92], shows the results that we intuitively expect from a join in a relational model that is capable to deal with uncertainty and ignorance.

[**Example 3.1**] Consider the following instance of two relations R_1 and R_2.

R_1

<u>*Z*</u>	*A*
z	0.4 [a_1] 0.6 [*]

R_2

<u>*A*</u>	*B*
a_1	0.7 [b_1] 0.3 [b_2]

Each relation consists of two attributes. Attributes *Z* and *A* are the keys of R_1 and R_2 respectively.
As argued in [BaGP 92], intuitively, the join between these relations on attribute *A* should result for (a_1, b_1) in probability[2] ranges between 0.28 and 0.7 and for (a_1, b_2) in probability ranges between 0.12 and 0.3. How to obtain these values was left as an open problem in [BaGP 92].
End of example 3.1

In order to obtain the desired values in Example 3.1, we describe in Section 3.3.1 a so-called conditional combination rule that is able to combine two bpa's. Then, in Section 3.3.2, we illustrate how this rule can be applied in order to solve the problem of information loss and to obtain the join result as discussed in Example 3.1.

3.3.1 Combining basic probability assignments (bpa's)

We start by discussing the combination rule of Dempster that is used to combine different bpa's [Shaf 76]. This rule will be referred as combination rule 1. As shown below, this rule cannot be applied straightforward in joining two relations in our extended relational model. Therefore, we adapt this rule, referred to as combination rule 2, to support joins in the extended model.

In the following a subset $S \subseteq D$ is called a focal element of a belief function *Bel* if m(*S*) > 0. Consider two belief functions Bel_x and Bel_y, with corresponding bpa's m_x and m_y, both defined over a set D. Let P_i, i = 1,2, …,p, and S_j, j= 1,2,..,q, be the focal elements of Bel_x and Bel_y, respectively. A graphical representation of both belief functions is given in Figure 3.1, in which the bpa's of the focal elements are depicted as segments of a line segment of length 1. In Figure 3.2, it is shown how the two bpa's can be orthogonally combined to obtain a square. The area of the total square is exactly 1. The area of a sub-square is the bpa assigned to the intersection of the focal elements P_i and S_j and can be computed by multiplying the values $m_x(P_i)$ and $m_y(S_j)$ respectively. A consequence of this procedure is that if the intersection of P_i and S_j is empty, it still obtains a bpa greater than zero. Since this is in contradiction with Def. 2.1, a normalization procedure should be applied as follows.
If the intersection of two focal elements is empty the sub-square is discarded and a value 0 is assigned to the area of this sub-square. The remaining areas of the sub-squares are adapted such that the sum of all sub-squares becomes one again.

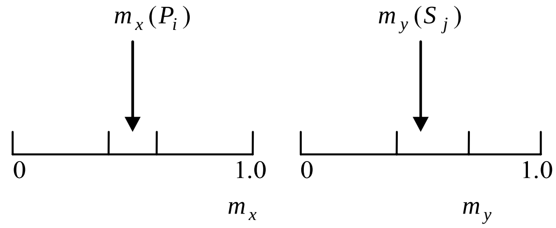

Figure 3.1: Graphical representation of two bpa's

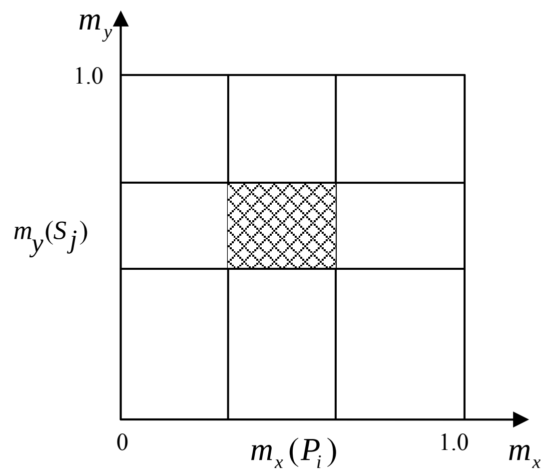

Figure 3.2: Combination of two bpa's

The following rule[3] provides the possibility to directly compute the combination of two bpa's

[**Combination Rule 1**] Let m_x and m_y be two bpa's defined over the same frame of discernment D. Then the combination of m_x and m_y results in

$$m_x \oplus m_y(Q) = \kappa^{-1} \sum_{\substack{i,j \\ P_i \cap S_j = Q}} m_x(P_i) m_y(S_j),$$ in which Q is a non empty subset of D and

$$\kappa = \sum_{\substack{i,j \\ P_i \cap S_j \neq \emptyset}} m_x(P_i) m_y(S_j).$$

We note that $\oplus$ symbolizes the combination operator, and κ is the normalization constant.

In order to support joins, this combination rule can not be applied in a straightforward way for two reasons. First, the combination rule is defined on a single frame of

---

[2] We note that probability is the term that is used in [BaGP92]

[3] This rule is known as the combination rule of Dempster in the literature.

discernment D and it makes sense to combine bodies of evidences if they pertain to the elements of frame D. In our extended relational model, we have bodies of evidences that pertain to different frames of discernment. In the context of example 3.1, the frame of discernment regarded to attribute A is $D_1 = \{a_1, a_2, a_3, ..., a_n\}$ and the frame of discernment regarded to attribute B is $D_2 = \{b_1, b_2\}$. In order to be able to apply the combination rule, the frames of discernment should be combined into a single frame. An alternative to obtain a single frame is to define a frame $D_{1,2} = \{(P, S) \mid P \subseteq D_1 \wedge S \subseteq D_2\}$. The second objection to apply combination rule 1 in a straightforward manner is that the rule assumes that bodies of evidences should be intuitively independent[4]. In our case it is quite obvious that we have to deal with dependency: in the context of Example 3.1, the value for b_1 is 0.7 if we observe that a_1 holds. In order to come up with a rule that is able to combine bodies of evidences that are dependent, we introduce the concept of conditional dependent bpa in the following definitions. On the basis of these definitions, we have formulated a new combination rule, referred to as combination rule 2.

[**Def 3.1**] Let D_x and D_y be two different frames of discernment, $P \subseteq D_x$, and m_x a bpa defined over D_x. A bpa defined over D_y is called conditionally dependent on m_x, denoted as $m_{y|x}(. \mid P)$, whenever the following holds: if $m_{y|x}(S \mid P) > 0$, in which $S \subseteq D_y$, then there exists a $P_i \subseteq D_x$ such that $P_i \cap P \neq \varnothing$ and $m_x(P_i \cap P) > 0$.

We note that *x* and *y* in $m_{y|x}(. \mid P)$ refer to the frames D_x and D_y and $m_{y|x}(S \mid P)$ is interpreted as the exact belief in *S* given (there is support for) *P*.

[**Def. 3.2**] A bpa $m_{y|x}(. \mid P_{fix})$ is conditionally dependent on m_x via a fixed set P_{fix} whenever the following holds $\sum_{S \subseteq D_y} m_{y|x}(S \mid P_{fix}) = 1$.

If we assume that the numbers in the relations of Example 3.1 represent bpa values, defined over $D_1 = \{a_1, a_2, a_3, ..., a_n\}$, then m_1 is: $m_1(a_1) = 0.4$ and $m_1(*) = 0.6$. Note that we neglect the role of key *Z* in R_1. In Section 3.3.2, Example 3.2, we give the rationale for this. The bpa defined over $D_2 = \{b_1, b_2\}$ is: $m_{2|1}(b_1 \mid a_1) = 0.7$ and $m_{2|1}(b_2 \mid a_1) = 0.3$. Since $\{a_1\} \subset D_1$, and $m_1(\{a_1\}) > 0$, $m_{2|1}(. \mid P)$ is conditionally dependent on m_1. Furthermore, $m_{2|1}(. \mid P)$ is also conditionally dependent on m_1 via set $\{a_1\}$ because $m_{2|1}(b_1 \mid a_1) + m_{2|1}(b_2 \mid a_1) = 1.0$.

According to the definition of conditional dependency, it only makes sense to talk about a bpa (and thus belief) in a set *S* given *P*, i.e., $m(S \mid P)$, if support for a non empty subset of *P* or a set equal to *P* is expressed in some bpa. In general, we conclude that if there is support for a non empty set P_j and $P_j \subseteq P$, then this means also support for P_j. We note that this is in agreement with our intuition.

In order to combine two bpa's m_x and $m_{y|x}(. \mid P)$, the combined bpa will pertain to a frame of discernment $D_{x,y}$ that is built up from D_x and D_y. Let us choose $D_{x,y} = \{(P, S) \mid P \subseteq D_x \wedge S \subseteq D_y\}$. We transform our bpa's m_x and $m_{y|x}(. \mid P)$ to the new frame of discernment $D_{x,y}$ by extending a focal element $P \subseteq D_x$ to a pair $(P,*)$ for m_x and a focal element $S \subseteq D_y$ to $(*, S \mid P)$ for $m_{y|x}(. \mid P)$.

Now, our running examples m_1 and $m_{2|1}(. \mid P)$ will become $\hat{m}_1(a_1,*) = 0.4$ and $\hat{m}_1(*,*) = 0.6$ and $\hat{m}_{2|1}(*, b_1 \mid a_1) = 0.7$ and $\hat{m}_{2|1}(*, b_2 \mid a_1) = 0.3$, respectively. The transformed bpa's are defined on the frame of discernment $D_{1,2} = \{(P, S) \mid P \subseteq D_1 \wedge S \subseteq D_2\}$.

The following proposition shows that a transformed bpa meets the properties of a bpa.

[**Proposition 1**] Let m_x be a bpa defined over a frame of discernment D_x. A transformation of the focal elements $P \subseteq D_x$ of m_x to $\hat{m}_x$ defined over pairs $(P,*)$, in which * represents ignorance pertaining to a different frame D_y, such that $\hat{m}_x(P,*) = m_x(P)$, yields: $\sum_{P \subseteq D_x} \hat{m}_x(P,*) = 1$ and $\hat{m}_x(\varnothing,*) = 0$.

[**Proof.**] The proof for this proposition is trivial and can be obtained by applying the Def. 2.1.

[4] Unfortunately, the theory of belief functions does not provide a hard definition about what independency precisely means.

Note that Proposition 1 also holds for a conditional bpa $m_{y|x}(.\,|\,P)$ that is transformed to $\hat{m}_{y|x}(*,.\,|\,P)$. Since we consider transformed bpa's in the following, we use *m* in stead of $\hat{m}$ to denote a transformed bpa.

To compute a combined bpa for two transformed bpa's, with regard to the frames D_x and D_y, m_x with focal element P_i, i = 1,2,…p, and $m_{y|x}(.\,|\,P_i)$ with focal elements S_j, j =1,2,..q, respectively, we hold an analogous reasoning as in the case of combination rule 1 to a certain extent. The figures 3.1 and 3.2 are substituted by Figure 3.3 and Figure 3.4, respectively. In Figure 3.4 the total area of the sub squares is one. Now, the area of a sub square is the bpa assigned to the combination of the focal elements $(P_i,*)$ and $(*,S_j\,|\,P)$ and can be computed by multiplying the values $m_X(P_i,*)$ and $m_{y|x}(*,S_j\,|\,P)$.

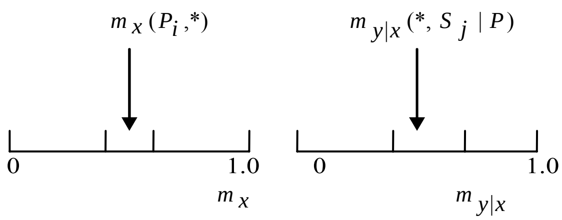

Figure 3.3: Graphical representation of two transformed bpa's

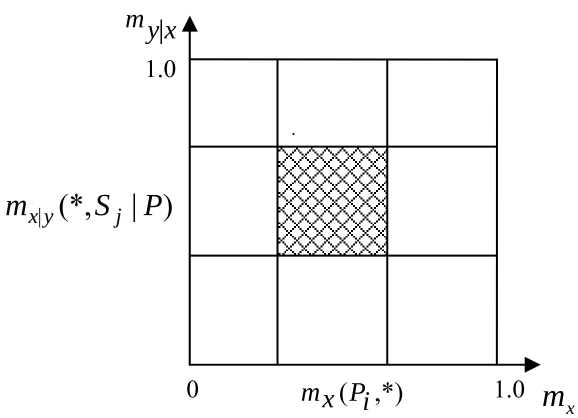

Figure 3.4: Graphical representation of the combination of a bpa with a conditional dependent bpa

The question that needs to be answered is: what is the meaning and result of the combination of the focal elements $(P_i,*)$ and $(*,S_j\,|\,P)$? The result of the combination of this two elements is either the pair (P_i,S_j) or $(P_i,*)$.

If $P \cap P_i \neq \varnothing$, then the result of the combination of $(P_i,*)$ and $(*,S_j\,|\,P)$ is the pair (P_i,S_j). In this case, we have support for set P_i which is expressed by means of m_x since $m_x(P_i,*) > 0$. Assume that *T* is the non empty set of the intersection between *P* and P_i, then $T \subseteq P$, and therefore we have support for *P*. Consequently, we may conclude support for set S_j, since $m_{y|x}(*,S_j\,|\,P) > 0$ and we have support for *P* via *T*. The contribution to the exact support for pair (P_i,S_j), i.e., $m_x \oplus m_{y|x}(P_i,S_j)$, on the basis of $m_x(P_i,*)$ and $m_{y|x}(*,S_j\,|\,P)$, is computed by multiplying the values $m_x(P_i,*)$ and $m_{y|x}(*,S_j\,|\,P)$. We note that $m_x \oplus m_{y|x}(P_i,S_j)$ is the size of the area of the shaded sub square in Figure 3.4.

If $P \cap P_i = \varnothing$, then the result of the combination of $(P_i,*)$ and $(*,S_j\,|\,P)$ is the pair $(P_i,*)$. Since the intersection between *P* and P_i results in the empty set, m_x does not have any focal elements that support set *P*, and therefore no statement can be made about the support for S_j. So, we have support for P_i due to m_x and no support for a specific subset of D_y on the basis of P_i. Therefore, we conclude the pair $(P_i,*)$ and the contribution to the exact support for this pair on the basis of $m_x(P_i,*)$ and $m_{y|x}(*,S_j\,|\,P)$ is again computed by multiplying those values.

This brings us to a combination rule, called combination rule 2, to combine a bpa with a conditional bpa. In this combination rule we sum up the sub squares that contribute to the bpa of a pair (P_i,S_j). We note that in contrary to combination rule 1, no normalization is required in combination rule 2, since the combination of two pairs $m_x(P_i,*)$ and $m_{y|x}(*,S_j\,|\,P)$ never leads to an empty set.

[**Combination Rule 2**] Let m_x be a bpa defined over a frame D_x and $m_{y|x}(*,.\,|\,P)$ a bpa that is conditionally dependent on m_x and defined over a frame D_y. Then, the combined bpa for a pair (P_i,S_j), in which $P_i \subseteq D_x$ and $S_j \subseteq D_y$, is computed as follows:

$$m_X \oplus m_{y|x}(P_i,S_j) = \begin{cases} \sum_{P_i \cap P \neq \varnothing} m_x(P_i,*)m_{y|x}(*,S_j\,|\,P) & \text{if } S_j \neq * \\ \sum_{P_i \cap P = \varnothing;\, S \subseteq D_y} m_X(P_i,*)m_{y|x}(*,S\,|\,P) & \text{else} \\ + \sum_{P_i \cap P \neq \varnothing} m_X(P_i,*)m_{y|x}(*,*\,|\,P) & \end{cases}$$

As discussed in the foregoing, the combination of $(P_i,*)$ and $(*,S_j \mid P)$ results in $(P_i,*)$ whenever $P \cap P_i = \varnothing$. In the case that $P \cap P_i \neq \varnothing$ the pair $(P_i,*)$ can be obtained due to a combination of the pairs $(P_i,*)$ and $(*,* \mid P)$. Therefore, the *else* part of combination rule consists of two expressions.

[**Proposition 2**] The combination of a bpa m_x defined over D_x and a bpa $m_{y|x}(*,. \mid P)$ that is conditionally dependent on m_x and defined over D_y results in a bpa that is defined on $D_{x,y} = \{(P_i,S_j) \mid P_i \subseteq D_x \wedge S_j \subseteq D_y\}$.

[**Proof.**] **T**he combination of the focal elements $(P_i,*)$ and $(*,S_j \mid P)$ corresponding to m_x and $m_{y|x}(*,. \mid P)$ respectively, results in either the pair (P_i,S_j) or $(P_i,*)$, in which $P_i \subseteq D_x$ and $S_j \subseteq D_y$. Since m_x and $m_{y|x}(*,. \mid P)$ are both bpa's, the following holds according to Proposition 1:
$\sum_{P_i \subseteq D_x} m_x(P_i,*) = 1$ and $\sum_{S_j \subseteq D_y; P \subseteq D_x} m_{y|x}(*,S_j \mid P) = 1$.

Combining m_x and $m_{y|x}(*,. \mid P)$ according to combination rule 2 results in a set of sub-squares. The sum of the areas of these sub-squares is exactly one. Therefore, $\sum_{(P_i,S_j) \subseteq (*,*)} m_x \oplus m_{y|x}(P_i,S_j) = 1$.
Since $m_x(\varnothing,*) = 0$ and $m_{y|x}(*,\varnothing \mid P) = 0$ is according to Proposition 1, the following holds $m_x \oplus m_{y|x}(P_i,\varnothing)$ and $m_x \oplus m_{y|x}(\varnothing,S_j)$.
End of proof

[**Proposition 3**] For the combination of a bpa m_x and a bpa $m_{y|x}(*,. \mid P_{fix})$ that is conditionally dependent on m_x via a fixed set P_{fix}, combination rule 2 reduces to

$$m_x \oplus m_{y|x}(P_i,S_j) = \begin{cases} m_x(P_i,*)m_{y|x}(*,S_j \mid P_{fix}) & \text{if } P_i \cap P_{fix} \neq \varnothing \\ m_x(P_i,*) & \text{else} \end{cases}$$

[**Proof.**] Both P_i and P_{fix} are fixed sets. So, the intersection between these two sets results in either a non empty set or an empty set. In the case of a non empty set between P_i and P_{fix}, combination rule 2 reduces to

$$m_x \oplus m_{y|x}(P_i,S_j) = \begin{cases} m_x(P_i,*)m_{y|x}(*,S_j \mid P_{fix}) & \text{if } S_j \neq * \\ m_x(P_i,*)m_{y|x}(*,* \mid P_{fix}) & \text{else} \end{cases}$$

which is equal to
$m_x \oplus m_{y|x}(P_i,S_j) = m_x(P_i,*)m_{y|x}(*,S_j \mid P_{fix})$. In the case of an empty set between P_i and P_{fix} combination rule 2 reduces to

$$m_x \oplus m_{y|x}(P_i,S_j) = \sum_{S \subseteq D_y} m_x(P_i,*)m_{y|x}(*,S \mid P_{fix})$$

Since $m_{y|x}(*,. \mid P_{fix})$ is conditionally dependent on m_x via a fixed set P_{fix}, the following holds according to definition 3.2: $\sum_{S \subseteq D_y} m_{y|x}(S \mid P_{fix}) = 1$.

Therefore,

$$m_x \oplus m_{y|x}(P_i,S_j) = \sum_{S \subseteq D_y} m_x(P_i,*)m_{y|x}(*,S \mid P_{fix}) =$$

$$m_x(P_i,*) \sum_{S \subseteq D_y} m_{y|x}(*,S \mid P_{fix}) = m_x(P_i,*).1 = m_x(P_i,*)$$

End of Proof.

To solve the joins in the next section, we use this specific version of combination rule 2.

[**Proposition 4**] The corresponding belief and plausibility functions of a bpa $m_x \oplus m_{y|x}$ defined over $D_{x,y} = \{(P_i,S_j) \mid P_i \subseteq D_x \wedge S_j \subseteq D_y\}$ are

$$Bel(P_i,S_j) = \sum_{P \subseteq P_i; S \subseteq S_j} m_x \oplus m_{y|x}(P,S) \text{ and}$$

$$Pl(P_i,S_j) = \sum_{\substack{P_i \cap P \neq \varnothing \\ S_j \cap S \neq \varnothing}} m_x \oplus m_{y|x}(P,S) \text{ respectively.}$$

[**Proof.**] The proof for this propositions can be obtained by applying the definitions of belief and plausibility functions.

3.3.2. Performing a join

So far, we extended the relational model by assigning bpa's to single attributes and we developed a rule to combine bpa's. In this subsection, we illustrate how a join is performed in our extended model. We restrict ourselves to equi-joins due to the page limitations. We start by elaborating on the value that a join attribute should assume after performing a join.

A traditional equi-join, is expressed by $R_1.A = R_2.A$, in which A is an attribute that appears in both relations R_1 and R_2. In this case, two tuples from the different relations are composed to a joined tuple if they have the same value for attribute *A*. Since in the extended model an attribute in R_1 as well as in R_2 may consist of a set of values, the question arises: what value attribute *A* should assume after a join?
Let A_1 and A_2 be the sets that contain the values for attribute *A* in relation R_1 and R_2 respectively. Then, both relations contains data that pertains to set $A_1 \cap A_2$.
.So, a joined tuple on the basis of *A* pertains to the set $A_1 \cap A_2$. Therefore, we define the set $A_1 \cap A_2$ as value for attribute *A* after a join.
In two subsequent examples, we illustrate how equi-joins should be performed in the extended model. A formal description of a join is beyond the scope of this paper.

[**Example 3.2**] We apply Proposition 3.2 to join the relations R_1 and R_2 as described in Example 3.1. Recall that we have the following transformed bpa for R_1: $m_1(a_1,*) = 0.4$ and $m_1(*,*) = 0.6$. For R_2 we have the conditional dependent bpa: $m_{2|1}(*,b_1 \mid a_1) = 0.7$ and $m_{2|1}(*,b_2 \mid a_1) = 0.3$. The domains of the attributes *A* and *B* are $D_1 = \{a_1, a_2, a_3, ..., a_n\}$ and $D_2 = \{b_1, b_2\}$.
Although it seems that the bpa on attribute *A* in R_1 is treated differently than the bpa on attribute *B* in R_2, this is not the case. Due to space limitations, we informally touch on this issue in this paper. Actually m_1 is conditionally dependent on the bpa of key *Z* via {z} (see example 3.1). Since there is no uncertainty about z and z holds, we can define the bpa on D_1 as an unconditional bpa m_1. Note that this reasoning does not hold for attribute *B* in R_2, since there is uncertainty about attribute *A* in R_1.
The combination of m_1 and $m_{2|1}(. \mid P \subseteq A)$ is sketched in Figure 3.5. On the horizontal and vertical axis the transformed bpa of m_1 and the conditional bpa $m_{2|1}(. \mid P \subseteq A)$ are depicted respectively. In Figure 3.5, for the sake of clarity, each subsquare contains the (new) combined pair of sets together with its corresponding bpa value. For example, the combination of the bpa values of the pairs $(a_1,*)$ (with value 0.4) and $(*,b_1 \mid a_1)$ (with value 0.7) results in a bpa of 0.4*0.7 = 0.28 for pair (a_1,b_1) (lower left subsquare in Figure 3.5). We note that the support for pair (a_1,b_1) is in line with our intuition. According to $m_{2|1}(*,b_1 \mid a_1) = 0.7$ there is support for b_1 whenever there is support for a_1. Since $m_1(a_1,*) > 0$ there is indeed support for a_1 and therefore support for b_1. A similar reasoning holds for the support of pair (a_1,b_2).
The combination of the bpa values of the pairs (*,*) and $(*,b_1 \mid a_1)$ results in a bpa of 0.7*0.6 =0.42 for pair $(*,b_1)$ for the following reason. The intersection between D_1 (=*) and $\{a_1\}$ is the set $\{a_1\}$. So, the combination of the pairs (*,*) and $(*,b_1 \mid a_1)$ is $(*,b_1)$. A similar reasoning holds for the support of pair $(*,b_2)$.

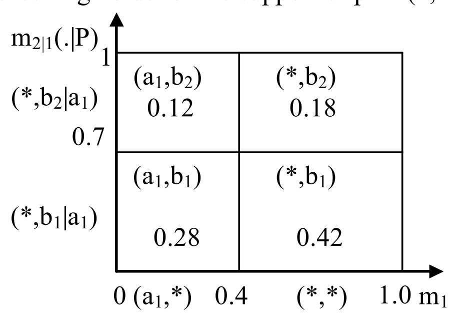

Figure 3.5: Graphical representation of the combination of m_1 and $m_{2|1}$

Consequently, the join between relations R_1 and R_2 results in the following relation.

Z	*A B*
z	0.28 $[a_1, b_1]$
	0.12 $[a_1, b_2]$
	0.42 $[*, b_1]$
	0.18 $[*, b_2]$

The corresponding *Bel* and *Pl* values are given below:

	Bel	*Pl*
$[a_1, b_1]$	0.28	0.7
$[a_1, b_2]$	0.12	0.3
$[*, b_1]$	0.7	0.7
$[*, b_2]$	0.3	0.3

The belief and plausibility values for attributes *A* and *B* are in line with the intuition as proposed in [BaPG 90]. Note, that in this example we have support for the set $\{a_1\}$ with a bpa value of 0.4 and support for the set $\{b_1\}$ with value 0.7, given that there is support for $\{a_1\}$. Therefore, we have a belief of 0.4*0.7 = 0.28 for the pair (a_1, b_1). However, it might be that the support for $\{a_1\}$ is 1.0, since a bpa value 0.6 is assigned to the set {*}, which contains the set $\{a_1\}$. Therefore, intuitively, the plausibility that the pair (a_1, b_1) may occur is 1.0*0.7 = 0.7.
End of Example 3.2

In the following we discuss a more complicated join between two relations. So far, suggested approaches were not able to perform this join in an adequate way.

[**Example 3.3**] Consider the snapshots of two relations called SHIP and DESC(ription).

SHIP

name	type
Maria	0.6 [Frigate] 0.3 [Tugboat] 0.1[*]

DESC

type	max-speed
Frigate	0.7 [20-knots] 0.3 [30-knots]
Tugboat	1.0 [15-knots]

The relation SHIP describes the type of a ship that an observed ship might be. For example, intelligence has been gathered in order to conclude that Maria may be either a Frigate with confidence 0.6 or a Tugboat with confidence 0.3, while some evidences leave us in doubt about the type of Maria. Therefore, 0.1 is assigned to all possible types of ships.

The relation DESC describes the maximal speed and the confidence in this speed under the condition that the type of a ship is known before-hand. So, the bpa assigned to the attribute max-speed is a conditional dependent on type..

Perhaps unnecessarily, we note that if we want to answer a question like "What is the maximum speed of Maria?", we have to perform a join between above mentioned relations. In order to compute this join between the relation SHIP and DESC on the attribute max-speed, we have to perform two times combination rule 2, namely a combination of the tuple of SHIP with the first tuple of DESC, and a combination of the tuple of SHIP with the second tuple of DESC.

The combination of the tuple (Maria {0.6 [Frigate], 0.3 [Tugboat], 0.1 [*]}) of SHIP with the tuple (Frigate, {0.7 [20-knots], 0.3 [30-knots]}) of DESC results in Figure 3.6a. On the horizontal axis the bpa of attribute type of relation SHIPS is depicted, called m_{SHIP}, and on the vertical axis the conditional bpa $m_{DESC| \ SHIP\text{-}FI}$ of attribute max-speed is depicted. For a similar reasoning as in Example 3.2, the bpa pertaining to relation SHIP is modelled as an unconditional bpa, while this is not the case for the bpa pertaining to DESC. The combination of the bpa values of the pairs (Tugboat, *) and (*, 30-knots | Frigate) results in a bpa value 0.3 *0.7 =0.21 for the pair (Tugboat,*). In this case, we have support for Tugboat, but this definitely does not mean support for Frigate, and therefore there is no support for a specific set of values of max-speed. For the combination of the bpa's of the remaining sets, a similar reasoning can be followed as in Example 3.2.

The combination of (Maria {0.6 [Frigate], 0.3 [Tugboat], 0.1 [*]}) of SHIP with the second tuple (Tugboat, {1.0 [15-knots]}) of DESC results in Figure 3.6b.

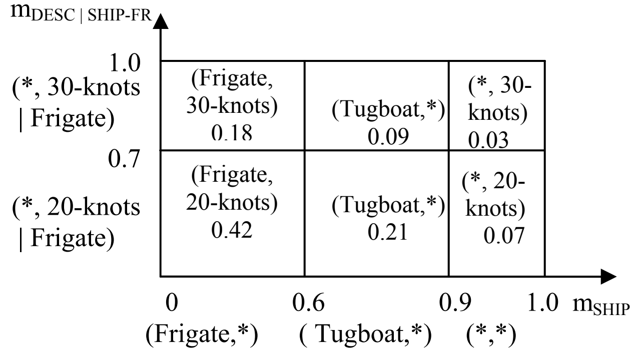

Figure 3.6a Combination of m_{SHIP} and $m_{DESC \mid SHIP\text{-}FR}$

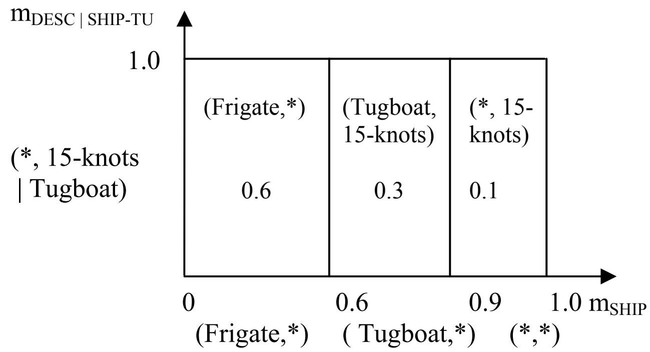

Figure 3.6b: Combination of m_{SHIP} and $m_{DESC \mid SHIP\text{-}TU}$

Eventually, the join of the relations SHIP and DESC on attribute type results in the following relation:

Name	type max-speed
Maria	0.42 [Frigate, 20-knots] 0.18 [Frigate, 30-knots] 0.3 [Tugboat, *] 0.07 [*, 20-knots] 00.3 [*, 30-knots]
Maria	0.6 [Frigate,*] 0.3 [Tugboat, 15-knots] 0.1 [*,15-knots]

The corresponding belief and plausibility values for the attributes type and max-speed are listed below.

	Bel	*Pl*
$m_{SHIP} \oplus m_{DESC \mid SHIP\text{-}FR}$		
[Frigate, 20-knots]	0.42	0.49
[Frigate, 30-knots]	0.18	0.21
[Tugboat,*]	0.3	0.4
[*, 20-knots]	0.49	0.49
[*, 30-knots]	0.21	0.21
$m_{SHIP} \oplus m_{DESC \mid SHIP\text{-}TU}$		
[Frigate,*]	0.6	0.7
[Tugboat, 15-knots]	0.3	0.4
[*,15-knots]	0.4	0.4

We note that the belief and plausibility values are in line with our intuition. For example, the belief of 0.42 that Maria is a Frigate and has a maximum speed of 20-knots can be understood by the fact that the bpa for a Frigate recorded in the relation SHIP is 0.6 and the bpa that the maximum speed is 20-knots for a Frigate is 0.7 (recorded in the relation DESC). The plausibility value of 0.49 for the same pair can be understood by the fact that a bpa of 0.1 is assigned to each possible subset of ships in relation SHIPS, implying ignorance. It might be the case that the bpa of 0.1 belongs to Frigate. Therefore, the plausibility that a ship is a Frigate with a maximum speed of 20 knots is (0.6 + 0.1)*0.7 = 0.49.
End of example 3.3

4. Conclusions & further research

Many researchers have pointed out that there is a need to extend the relational model with uncertainty and ignorance in order to support advanced database applications. In this paper, we build on on the pioneering work of [BaGP 90]. As in [BaGP 90], we extended the relational model in such a way that an uncertainty measure, called basic probability assignment (bpa), can be attached to an attribute. This leads to a revision of the definition of a tuple, now becoming a list of sets of attributes. To a certain extent, Dempster-Shafer theory is used to justify the theoretical aspects of our model. However, the properties of the theory appeared insufficient to support joins. Therefore, we came up with a so-called conditional combination rule, which can be to combine a bpa, m_1, with a bpa that is conditionally dependent on m_1. As has been shown, the application of the conditional combination rule solves the problem of information loss in joins. Furthermore, in our model we have a clear semantics of ignorance, which is adopted from Dempster-Shafer theory. A topic for further research is the formalization of the basic operators in the context of our model. The study of aggregation operators and nested operators is also a topic for further research. Furthermore, in the context of optimization our extended model gives cause for the study of a number of issues, such as the control of the complexity behavior of our combination rule, query optimization and so on.

References.

[BaGP 90] Barbara, D., Garcia-Molina, H., Porter, D., A Probabilistic Relational Data Model, in Proc. Int. Conference on Extending Database Technology, 1990, pp. 60-74.

[Barn 81] Barnett, J.A., Computational Methods for a Mathematical Theory of Evidence, in Proc. IJCAI-81, pp. 868-875 (1981).

[BhKa 86] Bhatnagar, R.K., Kanal, L.N., Handling Uncertain Information: A Review of Numeric and Non-numeric Methods, in Machine Intelligence and Pattern Recognition 4, Kanal, L.N., Lemmer, J.F., (eds), pp. 3-26, (1986).

[CaPi 87] Cavallo, R., Pittarelli, M., The Theory of Probabilistic Databases, in Proc. VLDB Int. Conf. on Very Large Databases, pp. 71-81.

[Demp 67] Dempster, A.P., Upper and Lower Probabilities Induced by a Multi-Valued Mapping, in Annals Math. Stat. 38, pp.325-339.

[Demp 68] Dempster, A.P., A Generalisation of Bayesian Inference, in Journal of the Royal Statistical Society, Series_B 30, pp. 205-247.

[DeSa 96] Dey, D., Sarkar, S., A Probabilistic Relational Model and Algebra . ACM TODS 21(3), 1996, pp. 339-369.

[Diac 78] Diaconis, P., Review of " A Mathematical theory of Evidence, in Journal of the American Statistical Society 73 (363), pp. 677-678, (1978).

[Fuhr 93] Fuhr, N., A Probabilistic Relational Model for the Integration of IR and Databases, ACM SIGIR 93, pp. 309-317.

[GuKT 91] Güntzer, U., Kießling, W., Thöne, H. New Directions for Uncertainty Reasoning in Deductive Databases. In Proc. ACM SIGMOD, Int Conf. on Management of Data, 1991, pp. 178-187

[GeHe 86] Gelenbe, E., Hebrail, G., A Probability Model of Uncertainty in Databases, in Proc. ICDE Int. Conf. on Data Engineering, 1986, pp. 328-333.

[GoSh 85] Gordon, J., Shortliffe, E.H., A Method for Managing Evidential Reasoning in a Hierarchical Hypothesis Space, in Artificial Intelligence 26, pp. 323-357, (1985).

[HaFa 92] Halpern, J.Y., Fagin, R., Two views of belief: belief as generalized probability and belief as evidence, in Artificial Intelligence 54, pp. 275-317.

[LaSa 94] Lakshmanan, L., Sadri, F., Modelling Uncertainty in Deductive Databases, in Proc. DEXA Int. Conf. on Databases, Expert Systems and Applications, 1994, 724-733.

[Lee 92] Lee, S-K, An Extended Relational Database Model for Uncertain and Imprecise Information, in Proc. VLDB, Int. Conf. on Very Large Databases, 1992, 211-220.

[RaMa 88] Raju, K, Majumdar, A. Fuzzy Functional Dependencies and Lossles Join Decomposition of Fuzzy Relational Database Systems, ACM TODS 13(2), 1988, 129-166.

[Shaf 76] Shafer, G., A Mathematical Theory of Evidence, Princeton University Press, Princeton 1976, 297p.

[ShLo 87] Shafer, G., Logan, R., Implementing Dempster's Rule for Hierarchical Evidence, in Artificial Intelligence 33, pp. 271-298, (1987).

[Wong 82] Wong, E., A Statistical Approach to Incomplete Information in Database Systems. ACM TODS 7(3), 1982.