Handling Uncertainty and Ignorance in Databases: A Rule to Combine Dependent Data

Sunil Choenni[1,2], Henk Ernst Blok[2], and Erik Leertouwer[1]

[1] Dutch Ministry of Justice, Research & Documentation Centre (WODC),
P.O.Box 20301, 2500 EH, The Hague, The Netherlands
{r.choenni, e.c.leertouwer}@minjus.nl
[2] University of Twente, Fac. of EEMCS, P.O. Box 217,
7500 AE, Enschede, The Netherlands
{r.s.choenni, h.e.blok}@utwente.nl

Abstract. In many applications, uncertainty and ignorance go hand in hand. Therefore, to deliver database support for effective decision making, an *integrated* view of uncertainty and ignorance should be taken. So far, most of the efforts attempted to capture uncertainty and ignorance with probability theory. In this paper, we discuss the weakness to capture ignorance with probability theory, and propose an approach inspired by the Dempster-Shafer theory to capture uncertainty and ignorance. Then, we present a rule to combine dependent data that are represented in different relations. Such a rule is required to perform joins in a consistent way. We illustrate that our rule is able to solve the so-called problem of information loss, which was considered as an open problem so far.

1 Introduction

Today, we distinguish several data models to represent and query data, such as the relational data model, object-oriented data models, XML data models, etc. Through the years a number of efforts has been devoted to capturing uncertainty in the context of relational databases [2,3,6,8,9,12,13,14,16]. Despite these efforts not all issues have been satisfactorily solved in the context of relational databases, while modelling uncertainty in other types of databases, such as XML databases is still in its childhood [1,7,10]. These approaches, except [13], are based on probability theory, and as a consequence they inherit the limitations of this theory. Probability theory is very suitable to capture uncertainty **but not** suitable to model ignorance. This has been noted and discussed in [2]. To overcome these limitations, Barbara et al. [2] introduced the so-called notion of missing probability, which is actually a way to model ignorance. However their approach suffers to a number of problems as will be illustrated in the next section.

Since uncertainty and ignorance go hand in hand in many applications, we feel that databases should support them in an integrated way. Suppose we have a document of which 80% is clearly visible and 20% of the document is damaged. This document

M.L. Lee, K.L. Tan, and V. Wuwongse (Eds.): DASFAA 2006, LNCS 3882, pp. 310 – 324, 2006.

contains an enormous amount of addresses, including addresses that give rise to suspicion. From the visible part, we can derive that 70% of the addresses is "normal" and 30% of them are considered as suspicious. So, if we have an arbitrary address A that comes from the visible part of the document, we know the distribution among normal and suspicious addresses, and therefore we are able to estimate whether A is a normal or a suspicious address. However, we will remain in *uncertainty* of the actual status of A, until we have checked in the document the details about A. For the damaged part of the document, we do not have any clue about the distribution of normal and suspicious addresses, therefore we are *ignorant* with regard to the addresses in this part of the document. If we want to estimate whether an arbitrary address B, of which it is unknown to what part of the document it belongs, is normal or suspicious, then we need to combine uncertainty and ignorance. Note, that estimating whether B is normal or suspicious on the basis of the distribution function that pertains only to the visible part will be unreliable. Therefore, to deliver database support for effective decision making, an *integrated* view of uncertainty and ignorance should be taken.

In this paper, we present how uncertainty and ignorance can be modelled in a relation, which consists of a set of tuples, and each tuple is a list of attribute values. Our approach is inspired by the Dempster-Shafer theory [5,11,15], but differs on main points of this theory (see Section 3). Then, we focus on how two relations, in which uncertainty and ignorance are captured can be combined in a consistent way to support joins in databases. We note that a join is an important operation to answer user queries posed on a relational database. The goal of this paper is to present the intuitive ideas behind our rule to combine dependent data and to show that we are able to solve the so-called problem of information loss (see Section 2), which was posed as an open problem in [2]. Therefore we will restrict ourselves in this paper to the combination of **two** relations. For the generalization of the rule to more than two relations we refer to a forthcoming paper and for the theoretical foundation of our model to capture uncertainty and ignorance in relational databases, we refer to [4].

The remainder of this paper is organised as follows. In Section 2, we discuss our problem definition in more detail and discuss why probability theory fails to solve the problem. Then, in Section 3, we briefly introduce our approach to model uncertainty and ignorance in databases. Then, in Section 4, we define our combination rule to combine dependent data represented in two different tables. In Section 5, we illustrate the application of our combination rule. Finally, Section 6 concludes the paper.

2 Problem Definition

In relational databases, a relation is defined over some attributes. An attribute takes a single value from a predefined domain. In our approach, we allow an attribute to take a *set* of values from a predefined domain **D**, and a function will be associated with this set, expressing the degree of uncertainty and ignorance among the elements in a set.

By means of the following example, which is similar to an example in [2], we introduce our problem definition in more detail. Suppose we want to predict the

planting behaviour of farmers. Therefore, we need to model some data about the weather and some data about the planting behaviour of farmers in the past. Let us assume that for the weather the possible outcome is either wet or dry. Now the KNMI (Royal Dutch Meteorological Institute) has 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 of 0.2. Since the probability of a dry and a wet season sum up to (0.6 + 0.2 =) 0.8, the remaining 0.2 actually implies ignorance with regard to the weather. In [2] , the authors model ignorance by assigning the probability of 0.2 to the set {wet, dry}. The semantic of this solution is that we do not make any statement how the probability of 0.2 is distributed among the elements of the set {wet, dry}. In the left table of Figure 1, the weather data is modelled. Furthermore, we have the following statistics for a dry season: 30% of the farmers planted turnips and 70% of them planted wheat if they expected a dry season. If farmers expected a wet season, they planted turnips. In the right table of Figure 1, we have modelled this data.

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

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

Fig. 1. Two base relations to model weather data

To gain insight in the planting behaviour of a farmer in the next season, the tables of Figure 1 need to be joined. To combine the probabilities, we may use the conditional rule of Bayes, namely, **Pr**(weather="w", plant="p") = **Pr**(plant="p" | weather="w")*__Pr__(weather ="w"), which results in Figure 2.

The first tuple in Figure 1 is telling us that the probability that it will be a dry season and a farmer will plant turnips is 0.18 and the probability that it will be a dry season and a farmer will plant wheat is 0.42. Note, the joined table contains answers to questions like: what is the probability that turnips/wheat will be planted next season?

source	weather plant
KNMI	(0.6*0.3 =) 0.18 [dry turnips] (0.6*0.7 =) 0.42 [dry wheat]
KNMI	(0.2 *1.0 =) 0.2 [wet turnips]

Fig. 2. Result of a join between the base relations depicted in Figure 1

As can be verified from Figure 2, ignorance (the probability of 0.2 assigned to {dry, wet}) has no influence on the join result. So, we have this information in one of our tables, but it is not used during the join, hence we have *information loss*.

From the above-mentioned example we observe the following. First of all, probability theory is not equipped to handle ignorance. For example, probability theory does not provide us the possibility to model the situation that 60% of the collected evidences points to a dry season and 20% to a wet season. Intuitively, we like to model this as **Pr**(dry) =0.6 and **Pr**(wet) = 0.2. However, this is in contradiction which one of the fundamental rules in probability theory. A corollary of the basic axioms of probability theory is the rule $\mathbf{Pr}(A) + \mathbf{Pr}(\neg A) = 1$. Let A represent the event "dry" season, thus $\mathbf{Pr}(A) = 0.6$. Actually, the probability of the event "wet" season is now determined and should be $\mathbf{Pr}(wet) = 1 - 0.6 = 0.4$ which is in contradiction with the collected evidences that are pointing to 0.2. Perhaps one might think that this problem can be solved by modelling an outcome space as Ω = {(wet), (dry), (wet, dry)} and defining a probability function p: Ω → [0,1]. In Appendix 1, we show that this does *not* lead to a solution.

Second, the approach proposed by Barbara et al. [2] leads to information loss and the embedding of their approach in probability theory is dubious, since it is in contradiction with the axioms of this theory (see also [4]).

Third, modelling ignorance by assigning a mass to a whole set of events, instead of (equally) distributing the mass among the elements of the set, is an attractive option and is pursued in this paper.

From the observations, we learn that ignorance and uncertainty are strongly intertwined. Therefore, for data management purpose, we need a theory in which these notions are embedded in an *integrated* way. In the next section, we propose our approach to capture uncertainty and ignorance.

3 Modelling Uncertainty and Ignorance in Databases

In this section, we start by introducing some basic notions from Dempster-Shafer theory [11] to capture uncertainty and ignorance in a single relation. However, to combine data from two different relations we need to extend the theory. We will discuss the extension in Section 3.2.

3.1 Basics of Dempster-Shafer Theory

We propose to attach a mass function, called **b**asic **p**robability **a**ssignment (bpa) to a set of attribute values in a relation. Based on this function, we will define the notion of ignorance.

[Def. 2.1] Let X be a set and $D_x = \{S \mid S \subseteq X\}$, then a function $m : D_x \rightarrow [0,1]$ is a bpa whenever $m(\varnothing) = 0$ and $\sum_{S \in D_x} m(S) = 1$.

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 $S \in D_x$ as $Bel(S) = \sum_{S' \subseteq S} m(S')$. Note, a bpa induces a belief function and conversely.

To define the notion of ignorance, we first define plausibility.

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

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

Now, we are able to model smoothly the data collected by KNMI in our example introduced in Section 2, without being in conflict with the axioms that belief functions should satisfy [15]. Note, for two sets S_1 and S_2, the following should hold for a belief function *Bel*: $Bel(S_1 \cup S_2) \geq Bel(S_1) + Bel(S_2) - Bel(S_1 \cap S_2)$.

Recall, in example 1, the KNMI collected evidences to predict whether it will be a dry or a wet season, and 60% of the evidences was pointing to a dry season, 20% to a wet season, and the remaining 20% of the evidences was neither pointing to a wet nor a dry season. This can be modelled as follows: *m*({dry}) = 0.6, *m*({wet}) =0.2, and *m*({dry, wet}) = 0.2. The corresponding belief function to *m* is: *Bel*({dry}) = 0.6, *Bel*({wet}) =0.2, and *Bel*({dry, wet}) = *m*({dry}) + *m*({wet}) + *m*({dry, wet}) = 1.0.

The plausibility for a dry season is: $Pl(\{\text{dry}\}) = 1 - Bel(\neg\{\text{dry}\}) = 1 - Bel'(\{\text{wet}\}) = 0.8$, and the ignorance with regard to a dry season is $Ig(\{\text{dry}\}) = Pl(\{\text{dry}\}) - Bel(\{\text{dry}\}) = 0.8 - 0.6 = 0.2$. We note that this is in line with our intuition, since 60% of the evidences are pointing to a dry season and 20% of the evidences leave us in ignorance because they are neither supporting a dry nor a wet season. So, an optimistic estimation for a dry season is 0.8. A similar reasoning can be hold for the prediction of a wet season.

3.2 Extending the Dempster-Shafer Theory

As can be seen from Figure 2, the combination of the base relations of Figure 1 leads to a relation , in which we like to obtain a bpa defined on a set that in turn consists to two distinct sets namely $D_{weather}$ ={dry, wet} and D_{plant} ={turnips, wheat}. Therefore, we need to extend the notion of bpa's to two distinct sets. Furthermore, the data in the weather table should be interpreted as that turnips will be planted with a bpa of 0.3, given the fact that it will be a dry season. So this means that the bpa defined on plant is dependent on the attribute weather. Therefore, we introduce the notion of dependent bpa. We start extending the Dempster-Shafer theory by defining a bpa on different sets.

[Def. 2.5] Let X and Y be two distinct sets and $D_x = \{S \mid S \subseteq X\}$ and $D_y = \{Q \mid Q \subseteq Y\}$. A function $m : D_x \times D_y \rightarrow [0,1]$ is a combined bpa on D_x and D_y whenever (1) $m(S,Q) = 0$, if $S = \varnothing$ or $Q = \varnothing$ and (2) $\sum_{S \in D_x} \sum_{Q \in D_y} m(S,Q) = 1$. A combined bpa will be denoted as c-bpa in the following.

Analogous to definitions 2.2 and 2.3, the belief and plausibility on D_x and D_y are defined as follows.

[Def. 2.6] Let *m* be a c-bpa defined over 2 distinct sets D_x and D_y, the belief in *(S,Q)*, in which $S \in D_x$ and $Q \in D_y$, is defined as $Bel(S,Q) = \sum_{S' \subseteq S} \sum_{Q' \subseteq Q} m(S', Q')$. The plausibility in the pair *(S,Q)* is defined as $Pl(S,Q) = 1 - Bel(\neg S, \neg Q)$.

[Def. 2.7] Let *m* be a c-bpa defined over two distinct sets D_x and D_y, and m_x a bpa defined on D_x, then *m* is called dependent on m_x whenever $\exists S \in D_x : \sum_{S_i \subseteq S} \sum_{Q \in D_y} m(S_i, Q) = 1$ and $m_x(S_i) > 0$. In the following we will denote a dependent bpa *m* on m_x as $m_{y|x}(. \mid S)$.

The intuition behind Def 2.7 is the following. The fact that a set $S \in D_x$ exists for which holds $\sum_{S_i \subseteq S} \sum_{Q \in D_y} m(S_i, Q) = 1$ means that we are talking about a "world" *S*, in which we distinguish all kinds of events *Q*, i.e., we reason about events *Q* given the world *S* or a subset of *S*. The fact $m_x(S_i) > 0$ implies that we have evidences that a world *S* exists, and therefore it is worthwhile to reason in world *S*.

Let us illustrate the notion of dependent bpa by means of our KNMI example, in which $D_{\text{weather}} = \{\text{dry}, \text{wet}\}$; and the bpa $m_{\text{weather}}(\{\text{dry}\}) = 0.6$, $m_{\text{weather}}(\{\text{wet}\}) = 0.2$ and $m_{\text{weather}}(\{\text{dry}, \text{wet}\}) = 0.2$.

Then, the c-bpa *m*(dry, turnips) = 0.7 and *m*(dry, wheat) = 0.3 is dependent on m_{weather} since *m*(dry, turnips) + *m*(dry, wheat) =1.0 and $m_{\text{weather}}(\{\text{dry}\}) > 0$. Note, in this case *S* = {dry}. If we formulate *m* as $m_{\text{plant}|\text{weather}}(\text{turnips} \mid \text{dry}) = 0.7$ and $m_{\text{plant}|\text{weather}}(\text{wheat} \mid \text{dry}) = 0.3$, then it is clear that it represents the first tuple in the right table of Figure 1.

The class of relational schemes that we consider in the remainder of this paper 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 relations introduced in Figure 1 are typically the base relations that we consider.

4 A Combination Rule

This section is devoted to the combination of a bpa m_x defined on a domain D_x and a dependent bpa, $m_{y|x}(. \mid S)$ defined on two domains D_x and D_y. In the following, a subset $S_i \subseteq S \in D_x$ or $Q_j \in D_y$ is called a focal element of a belief function if $m_x(S_i) > 0$ or $m_{y|x}(Q_j \mid S) > 0$. Consider two belief functions Bel_x and $Bel_{y|x}$, with

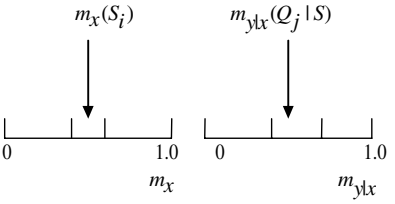

Fig. 3. Graphical representation of a bpa and a dependent bpa

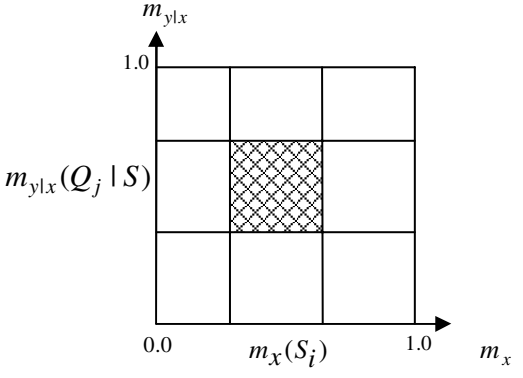

Fig. 4. Graphical representation of the combination of a bpa with a conditional bpa

corresponding bpa's m_x and $m_{y|x}(.|S)$. Let $S_i, i = 1,2,..., p$ and $Q_j, j = 1,2,..., q$ be the focal elements of Bel_x and $Bel_{y|x}$, respectively. A graphical representation of both belief functions is given in Figure 3, in which the bpa's of the focal elements are depicted as segments of a line segment of length 1.

In Figure 4, 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 rectangle is the c-bpa value assigned to the combination of the focal elements S_i and $Q_j \mid S$.

Let us focus on answering the following question: what is the meaning and result of the combination of the focal elements S_i and $Q_j \mid S$ in Figure 4? The result of the combination of these two elements is either the pair (S_i, Q_j) or $(S_i, *)$, in which * represents the whole domain D_y. We use the wildcard symbol, since the interpretation of the pair $(S_i, *)$ is that the first element is S_i, while the second element could be any subset of D_y. We distinguish three situations for providing an explanation for obtaining either the pair (S_i, Q_j) or $(S_i, *)$. In the following, the sets S_i and $Q_j \mid S$ are focal elements, and, as said before, $S_i \subseteq S \in D_x$ and $Q_j \in D_y$.

In the first situation, we assume $S \cap S_i = \varnothing$, then the result of the combination of S_i and $Q_j \mid S$ is the pair $(S_i, *)$. Since the intersection between S and S_i results in an empty set, m_x does not have any focal elements that supports set S, and therefore no

statement can be made about the support for Q_j. So, we have support for S_i due to m_x and no support for a specific subset of D_y on the basis of S_i. Therefore, we conclude the pair $(S_i,*)$ and the contribution to the exact support for this pair on the basis of $m_x(S_i)$ and $m_{y|x}(Q_j \mid S)$ is computed by multiplying those values. In the following $\oplus$ symbolizes the combination operator. We note that the value of $m_x \oplus m_{y|x}(S_i,*)$ is equal to the size of the area of the shaded rectangle in Figure 4.

Let us illustrate this situation by means of our running KNMI example. Suppose we want to combine $m_{\text{weather}}(\{\text{wet}\}) = 0.2$ and $m_{\text{plant}|\text{weather}}(\text{turnips} \mid \text{dry}) = 0.7$. Note, $m_{\text{weather}}(\{\text{wet}\}) = 0.2$ means that we have evidences for a wet season. However, $m_{\text{plant}|\text{weather}}(\text{turnips} \mid \text{dry}) = 0.7$ implies that there is support for planting turnips assuming that it will be dry. Since wet is in contradiction with dry, the conclusion should be that we have evidences for a wet season and no statement can be made about what to plant. Therefore, we conclude that the combination leads to $m_{weather} \oplus m_{plant|weather}(\{\text{wet}\},*) = 0.2 * 0.7 = 0.14$.

In the second situation, $S_i \subseteq S$, and therefore $S \cap S_i \neq \varnothing$. Then, the result of the combination of S_i and $Q_j \mid S$ is the pair (S_i, Q_j). In this case, we have support for set S_i which is expressed by means of m_x since $m_x(S_i) > 0$. Therefore, we have also support for set *S*, since *S* contains S_i. Consequently, we conclude support for Q_j. The contribution to the exact support for pair (S_i, Q_j), i.e., $m_x \oplus m_{y|x}(S_i, Q_j)$, on the basis of $m_x(S_i)$ and $m_{y|x}(Q_j \mid S)$, is again computed by multiplying the values $m_x(S_i)$ and $m_{y|x}(Q_j \mid S)$.

In the last situation, $S \cap S_i \neq \varnothing$ and $S_i \not\subseteq S$. Then, the result of the combination of S_i and $Q_j \mid S$ is the pair (S_i, Q_j) as well. Assume that *T* is the non empty set of the intersection between *S* and S_i. Since $m_x(S_i) > 0$ and we do not know anything about how this value is distributed among the elements or subsets of S_i, an option is to assign support to and to reason about *T*. Note, that this can be done for each subset of S_i that might be of interest. Since $T \subseteq S$ and we have support for *T,* this implies support for *S*. Consequently, we may conclude support for set Q_j, since $m_{y|x}(Q_j \mid S) > 0$ and we have support for *S* via *T*. Again, the bpa value for (S_i, Q_j) is computed as follows: $m_x \oplus m_{y|x}(S_i, Q_j) = m_x(S_i) m_{y|x}(Q_j, S)$.

We illustrate the last situation by means of our running example, where $m_{\text{weather}}(\{\text{dry}\}) = 0.6$, $m_{\text{weather}}(\{\text{wet}\}) = 0.2$, and $m_{\text{weather}}(\{\text{dry, wet}\}) = 0.2$. Consider the following two dependent bpa's: $m^1_{\text{plant}|\text{weather}}(\text{turnips} \mid \text{dry}) = 0.7$ and $m^1_{\text{plant}|\text{weather}}(\text{wheat} \mid \text{dry}) = 0.3$ and $m^2_{\text{plant} \mid \text{weather}}(\text{turnips} \mid \text{wet}) = 1.0$, representing the first and the second tuple in the right table of Figure 1 respectively.

Combining $m_{\text{weather}}(\{\text{dry, wet}\}) = 0.2$ with $m^1_{\text{plant|weather}}$ results into: $m_{\text{weather}} \oplus m^1_{\text{plant|weather}}(\text{dry, turnips}) = 0.2*0.7 = 0.14$ and $m_{\text{weather}} \oplus m^1_{\text{plant|weather}}(\text{dry, wheat}) = 0.2*0.3 = 0.06$. We note that $m_{\text{weather}}(\{\text{dry, wet}\}) = 0.2$ implies support for the set {dry, wet}. However we do not know how the mass of 0.2 is distributed among the elements of {dry, wet}. Since for $m^1_{\text{plant|weather}}$, a dry weather is of interest, a possible distribution is the computed $m_{\text{weather}} \oplus m^1_{\text{plant|weather}}$. In $m^2_{\text{plant|weather}}$, a wet weather is of interest. Then, the combination of $m_{\text{weather}}(\{\text{dry, wet}\}) = 0.2$ with $m^2_{\text{plant|weather}}$ results in $m_{\text{weather}} \oplus m^2_{\text{plant|weather}}(\text{wet, turnips}) = 0.2$. So, the combination of $m_{\text{weather}}(\{\text{dry, wet}\}) = 0.2$ may result in different possible distributions depending on the set of weather that is of interest for a dependent bpa. This is in line with our intuition.

Let us formulate now our combination rule, in which sum up the rectangles that contribute to the bpa of a pair (S_i, Q_j).

$$m_x \oplus m_{y|x}(S_i, Q_j) = \begin{cases} \sum_{S_i \cap S \neq \varnothing} m_x(S_i) m_{y|x}(Q_j \mid S) & \text{if } Q_j \neq * \\ \sum_{\substack{S_i \cap S = \varnothing \\ Q \in D_y}} m_x(S_i) m_{y|x}(Q \mid S) + \sum_{S_i \cap S \neq \varnothing} m_x(S_i) m_{y|x}(* \mid S) & \text{else} \end{cases} \quad (1)$$

As discussed in the foregoing, the combination of S_i and $Q_j \mid S$ results in $(S_i, *)$ whenever $S_i \cap S = \varnothing$. In the case, $S_i \cap S \neq \varnothing$ the pair $(S_i, *)$ can be obtained due to a combination of S_i and $* \mid S$ as well. Therefore, the *else* part of combination rule consists of two expressions.

The following proposition considers a special case of our combination rule. As will be illustrated in the next section, it appears that this special case is sufficient to solve the open problem posed in [2].

[Prop. 1] Let m_x be a bpa defined over D_x, and S_{fix} be a fixed set in a dependent c-bpa $m_{y|x}(. \mid S_{fix})$, which is defined over D_x and D_y. Then equation (1) reduces to

$$m_x \oplus m_{y|x}(S_i, Q_j) = \begin{cases} m_x(S_i) m_{y|x}(Q_j \mid S_{fix}) & \text{if } S_i \cap S_{fix} \neq \varnothing \\ m_x(S_i) & \text{else} \end{cases} \quad (2)$$

[*Proof.*] The intersection of S_i and S_{fix} is either empty or not empty. If the intersection between S_i and S_{fix} results in a non empty set then our combination rule, i.e., equation (1) reduces to

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

which is equal to $m_x \oplus m_{y|x}(S_i, Q_j) = m_x(S_i) m_{y|x}(Q_j \mid S_{fix})$.

If the intersection between S_i and S_{fix} results in an empty set, then our combination rule, i.e., equation (1) reduces to $m_x \oplus m_{y|x}(S_i, Q_j) = \sum_{Q \in D_y} m_x(S_i) m_{y|x}(Q \mid S_{fix})$. According to Def 2.7, $\sum_{Q \in D_y} m_{y|x}(Q \mid S_{fix}) = 1$ since $m_{y|x}(. \mid S_{fix})$ is a dependent c-bpa.

Therefore,

$$m_x \oplus m_{y|x}(S_i, Q_j) = \sum_{Q \in D_y} m_x(S_i) m_{y|x}(Q \mid S_{fix}) = m_x(S_i) \sum_{Q \in D_y} m_{y|x}(Q \mid S_{fix}) = m_x(S_i) . \square$$

5 Illustrative Examples

In this section, we illustrate how our combination rule can be applied to support a join in relational databases. As noted before, a join is an important operator and combines data that is stored in different relations. We restrict ourselves to equi-joins due to the page limitations. Example 5.1 in this section is literally adopted from [2]. This example was posed as an open problem by its authors. We will illustrate how this problem can be solved by applying our combination rule. 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 our extended relational 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, the set $A_1 \cap A_2$ contains data that can be found in both relations. So, a joined tuple on the basis of *A* pertains to the set $A_1 \cap A_2$. Therefore, we define the value for attribute *A* after a join as the set $A_1 \cap A_2$. By means of examples, we illustrate how our combination rule can be applied in performing joins. The following example, adopted from [2], shows the results that we intuitively expect from a join in a relational model that is capable to deal with uncertainty and ignorance.

[Example 5.1] Consider the following instances of two relations R_1 and R_2.

R_1

Z	A
z	0.4 [a_1]
	0.6 [*]

R_2

A	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 [2], intuitively, the join between these relations on attribute *A* should result for pair (a_1, b_1) in probability[1] range between 0.28 and 0.7 and for pair (a_1, b_2) in probability range between 0.12 and 0.3. How to obtain these values was left as an open problem in [2].

In the next example, we illustrate how we can obtain these desired values by applying proposition 1, which is a special case of our combination rule. Then, in Example 5.3 we discuss a more complicated case.

[Example 5.2] From the relations of example 5.1, it is clear that we have the following bpa defined on attribute A in relation R_1: $m_1(\{a_1\}) = 0.4$ and $m_1(*) = 0.6$. Recall that * represents the whole domain of an attribute. In relation R_2, we have a c-bpa defined over the attributes *A* and *B* and which is dependent on m_1. This bpa looks as follows: $m_{2|1}(\{b_1\} \mid \{a_1\}) = 0.7$ and $m_{2|1}(\{b_2\} \mid \{a_1\}) = 0.3$. From now on we omit the brackets for a set, if it is clear that we are dealing with a set. 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 dependent on the bpa of key *Z* via {z}. Since there is no uncertainty about z and no other relations contains attribute Z, we can define the bpa on *A* in R_1 as an independent 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 $m_{2|1}(. \mid S)$, in which *S* is a subset of or equal to the domain of attribute *A*, is sketched in Figure 5. On the horizontal and vertical axis the bpa m_1 and the dependent bpa $m_{2|1}(. \mid S)$ are depicted respectively. We note that here the set *S* is a fixed set that consists of $\{a_1\}$, and therefore Proposition 1 is applied.

In Figure 5, for the sake of clarity, each rectangle contains the (new) combined pair of sets together with its corresponding bpa value. For example, the combination of the bpa values of the elements (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 rectangle in Figure 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 there is support for b_1. A similar reasoning holds for the support of pair (a_1, b_2).

The combination of the bpa values of (*) 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 {*} and $\{a_1\}$ is the set $\{a_1\}$. So, the combination of the elements (*) and $(b_1 \mid a_1)$ is $(*, b_1)$. Note that $m_1(*) = 0.6$ means that there is support for the whole domain of attribute *A*.

[1] We note that probability is the term that is used by the authors in [2].

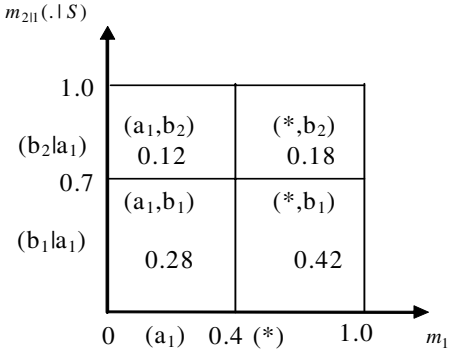

Fig. 5. Graphical representation of the combination of m_1 and $m_{2||}(.|S)$

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

(a)

	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

(b)

Fig. 6. (a) Join result between R_1 and R_2 and (b) Corresponding belief and plausibility values

However, no statement can be made about the distribution of 0.6 among the subsets of the domain of A. Since now the subset $\{a_1\}$ is of interest, we consider this set as option. A similar reasoning holds for the support of pair $(*, b_2)$.

Consequently, the join between relations R_1 and R_2 is given in Fig. 6(a) and the corresponding belief and plausibility values for the different pairs are given in Fig 6(b).

The belief and plausibility values for attributes *A* and *B* are in line with the intuition as proposed in [2]. 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 pair (a_1, b_1) may occur is 1.0*0.7 = 0.7.

[**Example 5.3]** Consider the snapshots of two relations called ship and description.

ship

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

description

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 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 description 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 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.

To compute a join between the relations ship and description on the attribute max-speed, we have to perform two combinations, namely a combination of the tuple of ship with the first tuple of description, and a combination of the tuple of ship with the second tuple of description.

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 description results in the left part of Figure 7. Note, Frigate is a fixed set in the left part of Figure 7, while in the right part of Figure 7 Tugboat is a fixed set. On the horizontal axis the bpa of attribute type of relation ship is depicted, called m_{ship}, and on the vertical axis the dependent bpa $m_{\text{desc}\,|\,\text{ship-Fr}}$ corresponding to the first tuple of relation description is depicted. For a similar reasoning as in Example 3.2, the bpa pertaining to relation ship is modeled as an independent bpa. 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,*) (lower middle rectangle in the right part of Figure 7). 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 relation ship with the second tuple (Tugboat, {1.0 [15-knots]}) of relation description results in the right part of Figure 7.

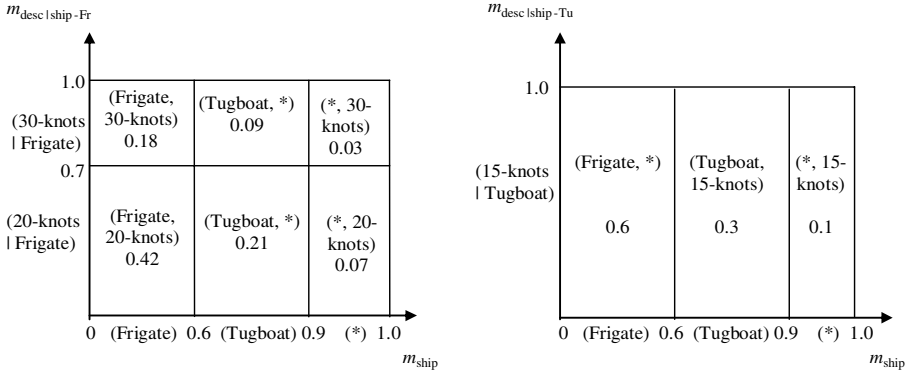

Fig. 7. Graphical representation of the combination of the bpa's corresponding to ship and description

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]

(a)

	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

(b)

Fig. 8. (a) Join result between ship and description and (b) Corresponding Bel and Pl values

The result of the join between the relations ship and description together with the corresponding belief and plausibility values is given in Figure 8.

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 description). 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 ship, 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.

6 Conclusion and Further Research

Many researchers have pointed out that there is a need to handle uncertainty and ignorance in database applications. Most of the efforts applied probability theory to capture uncertainty and ignorance. As has been argued in Section 2, probability theory is suitable to capture uncertainty but *not* to capture ignorance. In this paper, we have proposed a framework to capture uncertainty and ignorance in an integrated way. Although our framework can be tailored to different type of data models, we elaborate it for the relational model. We assume that an attribute can assume a set of values instead of a single value. And we assign, inspired by the Dempster-Shafer theory [5,11,15], a so-called basic probability assignment (bpa) to an attribute. However, the properties of the Dempster-Shafer theory appeared insufficient to support joins. Therefore, we extended the theory with the notion of a "dependent" bpa. Such a bpa provides us the possibility to take dependencies between data into account. Based on the notion of dependent bpa, we came up with a combination rule to combine a bpa, m_1, with a bpa that is dependent on m_1. As has been shown, the application of this combination rule solves the problem of information loss that occurs as a consequence of joins. Until now, the problem of information loss was posed as an open problem in the literature [2]. Furthermore, in our model we have a clear semantics of ignorance.

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

1. Al-Khalifa, S., Yu, C., Jagadish, H.V., Querying Structured Text in XML Database, Int. Conf. ACM SIGMOD 2003.
2. Barbara, D., Garcia-Molina, H., Porter, D., A Probabilistic Relational Data Model, in Proc. Int. Conference on Extending Database Technology, 1990, pp. 60-74.
3. Cavallo, R., Pittarelli, M., The Theory of Probabilistic Databases, iProc. VLDB Int. Conf. on Very Large Databases 1987
4. Choenni, R., Blok, H.E., Fokkinga, M., Extending the Relational Model with Uncertainty and Ignorance, Technical Report, University of Twente.
5. Dempster, A.P., Upper and Lower Probabilities Induced by a Multi-Valued Mapping, in Annals Math. Stat. 38, pp.325-339.
6. Dey, D., Sarkar, S., A Probabilistic Relational Model and Algebra . ACM TODS 21(3), 1996, pp. 339-369.
7. Fuhr, N., A Probabilistic Relational Model for the Integration of IR and Databases, ACM SIGIR 93, pp. 309-317.
8. 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
9. Gelenbe, E., Hebrail, G., A Probability Model of Uncertainty in Databases, in Proc. ICDE Int. Conf. on Data Engineering, 1986, pp. 328-333.
10. Hung, E., Getoor, L., Subrahmaniam, PXML: A Probabilistic Semi-structured Data Model and Algebra, Int. Conf. on Data Engineering 2003.
11. Halpern, J.Y., Fagin, R., Two views of belief: belief as generalized probability and belief as evidence, in Artificial Intelligence 54, pp. 275-317.
12. Lakshmanan, L., Sadri, F., Modelling Uncertainty in Deductive Databases, in Proc. Databases, Expert Systems and Applications, 1994.
13. 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.
14. Raju, K, Majumdar, A. Fuzzy Functional Dependencies and Lossles Join Decomposition of Fuzzy Relational Database Systems, ACM TODS 13(2), 1988, 129-166.
15. Shafer, G., A Mathematical Theory of Evidence, Princeton University Press, Princeton 1976, 297p.
16. Wong, E., A Statistical Approach to Incomplete Information in Database Systems. ACM TODS 7(3), 1982.

Appendix

Perhaps one might think that the KNMI problem of Section 2 can be solved by probability theory by choosing a suitable model for the outcome space. One could argue to choose the outcome space as follows $\Omega = \{[\text{wet}], [\text{dry}], [\text{wet, dry}]\}$. Then we can use probability theory to reason about this space. We define now a probability function $p: \Omega \rightarrow [0,1]$. Let say $p([\text{dry}]) = 0.6$, $p([\text{wet}]) = 0.2$, and $p([\text{dry, wet}]) = 0.2$. If we compute the probability of the union of wet and dry, then $p([\text{dry}] \cup [\text{wet}]) = p([\text{dry}]) + p([\text{wet}]) = 0.8$, which is in contradiction with $p([\text{dry, wet}]) = 0.2$.