Databases for Interval Probabilities

Wenzhong Zhao*, Alex Dekhtyar†, and Judy Goldsmith‡
Department of Computer Science, University of Kentucky, Lexington, KY 40503, USA

We present a database framework for the efficient storage and manipulation of interval probability distributions and their associated information. While work on interval probabilities and on probabilistic databases has appeared before, ours is the first to combine these into a coherent and mathematically sound framework including both standard relational queries and queries based on probability theory. In particular, our query algebra allows users not only to query existing interval probability distributions, but also to construct new ones by means of conditionalization and marginalization, as well as other more common database operations.

1 INTRODUCTION

A probability distribution represents the likelihoods of each of a set of possible events. However, there are times when we are unsure about those likelihoods, either because our information about the world is unreliable or because it is incomplete. One way to represent such uncertainty is by using probability intervals[8,21].

Interval probabilities may arise in a specific application in many ways. While small numbers of such distributions can be handled ad-hoc, with the increase of application domain sizes, the amount of information eventually overwhelms ad-hoc methods. Our work concentrates on the problem of efficiently and correctly managing large quantities of interval probability distributions.

The hypothetical example we use (Section 2) is about an election in a town of Sunny Hill. To everyone's surprise, the Rhinoceros Party has won the senate seat and swept local elections. However, a referendum that the Rhinoceros Party supported, to legalize AI conferences, has failed. Pundits and lawyers wish to investigate the fact that the usual indicators, polls, yard signs, etc., created expectations that were not fulfilled. The data on which such analyses are based are probability tables of various forms. To be useful, each table must be clearly labeled by its origin, format, and any implicit conditions, such as "From a poll of Elephant men at their annual pig roast and fund raiser."

In this work, we provide a flexible database framework for storing and managing such labeled interval probability distributions. The data model for our framework, Extended Semistructured Probabilistic Objects (ESPOs) (Section 3) allows us to

* Author to whom all correspondence should be addressed: e-mail: wzhao0@cs.uky.edu
† e-mail: dekhtyar@cs.uky.edu
‡ e-mail: goldsmit@cs.uky.edu

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. X, XX-XX (2004)
 Published online in Wiley InterScience (www.interscience.wiley.com). · DOI 10.1002/int.10143

store a wide variety of interval probability distributions and related information. The querying mechanism for the ESPO model, ESP-Algebra (Section 5), provides operations for accessing and manipulating this data.

In order to make sense of the operations on interval probability distributions, we must fix a semantics. We choose the *possible worlds semantics*[8,9,15,23]. This semantics (Section 4) captures the idea that, while exact point probabilities distributions are not known, they are known to lie within given intervals. There is a growing literature on interval probability distributions (see, for example,[2,3,6,8,14,16,23] and[21,22].) The work cited above, however, does not address the problem of efficiently and correctly managing collections of such distributions, which is the main contributions of this paper. There has been some work on databases of probability distributions, and of interval probabilities, but ours is the first to offer a query algebra specifically designed to store and manipulate interval probability tables as the primary data objects. (Discussion of other database management systems that use interval probabilities can be found in Section 6.)

2 TROUBLE IN SUNNY HILL

In order to illustrate our motivation we offer some data about the Sunny Hill election. It is understood that many people misrepresent the truth to pollsters. Therefore, poll data, such as the collection shown in Figure 1, is assumed to have a margin of error. A typical statement is, *"The straight Donkey ticket for the Senate and mayoral election is preferred by 33% of respondents +/- 3%"* (See Poll1 table from Figure 1). Although the actual polling data indicates statistical information about respondents, it can be interpreted probabilistically as *"The probability that a resident of Sunny Hills will vote straight Donkey ticket in the elections is between 29% and 36% based on the October 18 poll."* (See the top line of Poll1 table in Figure 1.)

Given a database of polling data and the desire for a particular set of probabilities, how can the pundits and lawyers access the information? Typically, polling data is stored in a raw format by polling organizations, often in a relational DBMS, and is analyzed using a variety of statistical and/or mathematical packages, such as SAS, SPSS or MatLab. This software can be used to construct probability distributions such as those shown in Figure 1, and to perform other manipulations of the data. However, neither traditional relational DBMS nor statistical software deal with storage and retrieval of the probability tables constructed during the analysis. Our DBMS allows the pundits and lawyers to answer questions such as:

- Find all probability distributions for voters from Downtown based on the surveys taken within two weeks of the election date;
- Find the distributions of mayoral vote for voters who plan to vote for building a new park;
- Find all distributions in which the Donkey mayoral candidate receives more than 40% of votes.

In order to answer such questions, our query language must be able to manipulate the probability distributions stored in the database and perform transformations of the distributions according to the laws of probability theory. For example, the second query above, applied to a joint distribution of votes for mayoral race and two ballot initiatives (such as Poll2 in Figure 1), should result in the computation of

id: Poll1
population: entire town
date: October 18

senate	mayor	l	u
Donkey	Donkey	29%	36%
Donkey	Elephant	5%	10%
Donkey	Rhino	5%	12%
Elephant	Donkey	7%	14%
Elephant	Elephant	25%	34%
Elephant	Rhino	6%	13%
Rhino	Donkey	4%	9%
Rhino	Elephant	3%	8%
Rhino	Rhino	8%	17%

id: Poll2
population: Donkey men
date: October 26
senate vote: Donkey

mayor	park	legalization	l	u
Donkey	yes	yes	44%	52%
Donkey	yes	no	12%	16%
Donkey	no	yes	8%	12%
Donkey	no	no	4%	8%
Elephant	yes	yes	5%	10%
Elephant	yes	no	1%	2%
Elephant	no	yes	3%	4%
Elephant	no	no	6%	8%
Rhino	yes	yes	2%	4%
Rhino	yes	no	1%	3%
Rhino	no	yes	3%	5%
Rhino	no	no	1%	4%

id: Poll3
population: entire town
date: October 22
senate vote: Donkey
mayor vote: Rhino

park	legalization	l	u
yes	yes	56%	62%
yes	no	14%	20%
no	yes	21%	25%
no	no	3%	7%

id: Poll4
population: South Side
date: October 12
sample size: 323

mayor	l	u
Donkey	20%	26%
Elephant	42%	49%
Rhino	25%	33%

id: Poll5
population: Downtown
date: October 12
sample size: 275

mayor	l	u
Donkey	48%	55%
Elephant	25%	30%
Rhino	20%	24%

Fig. 1. Polling Data for Sunny Hills elections.

a marginal probability distribution for the mayoral vote and the park ballot initiative (by excluding the second initiative from the distribution) and subsequent conditioning on park=yes.

This example indicates the importance of the following features: (i) probability distributions and their associated, non-probabilistic information are treated as single complex objects; (ii) probability distributions with different structures (e.g., different number/type of random variables involved) are stored and accessed together; (iii) query language facilities are provided for retrieval of full distributions based on their properties, and for retrieval of parts of distributions (individual rows of the probability tables); (iv) query language facilities are provided for manipulations and transformations of probability distributions according to the laws of probability theory; (v) *interval* probability distributions are correctly handled.

In this paper, we describe how Extended Semistructured Probabilistic Object (ESPO) model and ESP-Algebra achieve these properties.

3 EXTENDED SEMISTRUCTURED PROBABILISTIC OBJECT (ESPO) DATA MODEL

Semistructured Probabilistic Objects (SPOs) were introduced by Dekhtyar, Goldsmith and Hawkes[10]. Each SPO contains a *probability table* of *participating random variables*, along with *conditionals*, the given conditions on which the probabilities in the table were conditioned, and *context*, the additional information about known values of parameters which were not considered by the application to be random variables.

The extended semistructured probabilistic objects (ESPOs) differ from SPOs in several ways. Most important is that probabilities are given as intervals rather than exact values. In addition, we allow context and conditionals to be associated with subsets of the participating variables. Finally, we include a *path*, an indicator of the origin of an ESPO. Figure 2 shows the anatomy of ESPOs.

ω: S ← *path expression*

date: October 23
gender: male ← *extended context*
respondents: 238, {senate}
respondents: 195, {legalization}
overlap: 184

← *random variables*

senate	legalization	l	u
Rhino	yes	0.04	0.11
Rhino	no	0.1	0.15
Donkey	yes	0.22	0.27
Donkey	no	0.09	0.16
Elephant	yes	0.05	0.13
Elephant	no	0.21	0.26

← *interval probability table*

mayor: Donkey{senate, legalization} ← *extended conditional*

Fig. 2. Extended Semistructured Probabilistic Object

DEFINITION 1. *Let $\mathcal{R} = \{A_1, \ldots, A_k\}$ be a set of relational attributes and $\mathcal{V} = \{v_1, \ldots, v_N\}$ a universe of random variables. Let $V \subseteq \mathcal{V}$ be a set of participating random variables.*
A context entry *is a pair (A, a), where $A \in \mathcal{R}$ and $a \in dom(A)$. An* extended context entry *over $\mathcal{R}$ is a triple (A, a, W), where $A \in \mathcal{R}$, $a \in dom(A)$ and $W \subseteq V$. A* conditional *is a pair (u, X), where $u \notin V$ and $X \subseteq dom(u)$. An* extended conditional *is a triple (u, X, W), where $u \notin V$, $X \subseteq dom(u)$ and $W \subseteq V$.*

The set $\mathcal{R}$ of relational attributes contains the non-stochastic variables present in an application domain. A context entry is simply an assignment of a value to a context attribute. An extended context entry associates a set of random variables with such an assignment. Similarly, a conditional entry specifies a value or a set of values for the specific random variable. An extended conditional associates a set of random variables with such conditioning information.

In addition to extending context and conditionals and switching to interval probabilities, we also define a *path* for an ESPO. The path of an ESPO S indicates its origin. When an ESPO is inserted into a database it gets a unique id as its path. If S is the result of a sequence of query algebra operations, the construction of S is documented in its path. The exact syntax on construction of paths is explained in Section 5.

We can now give the definition of an Extended SPO (ESPO).

DEFINITION 2. *Let $\mathcal{P}$ = C[0,1] be a set of all subintervals of the interval $[0, 1]$. An* ***Extended Semistructured Probabilistic Object (ESPO)*** *S is a tuple $S = \langle T^+, V, P, C^+, \omega\rangle$, where*

- *$T^+ = \{(A, a, W)\}$ is a set of extended context entries;*

• $V = \{v_1, \ldots, v_q\} \subseteq \mathcal{V}$ *is a set of random variables that* participate *in* S. *We require that* $V \neq \emptyset$*;*

• $P : dom(V) \longrightarrow \mathcal{P}$ *is the* probability table *of* S. P *must be* consistent *(see Definition 5 in Section 4);*

• $C^+ = \{(u, X, W)\}$ *is a set of extended conditionals, such that* $(\forall u)((u, X, W) \in C^+ \rightarrow u \notin V)$*;*

• ω*, called a* path *of* S*, is an expression of the Extended Semistructured Probabilistic Algebra.*

4 SEMANTICS FOR INTERVAL PROBABILITIES

Earlier work on Semistructured Probabilistic Objects used *point probabilities*[10].

In this paper we assume that the probability space is $\mathcal{P}$ = C[0,1], the set of all subintervals of the interval $[0,1]$. This section formally introduces the possible worlds semantics for probability distributions over $\mathcal{P}$ and the notions of *consistency* and *tightness* for probability distributions. While there exist a variety of approaches to interpreting interval probabilities[21] the possible worlds approach has been adopted by a number of researchers as the one that admits direct computations. In particular, the semantics described here follows the work of de Campos, Huete and Moral[8] and is similar to the work of Weichselberger[23]. In the context of databases, this semantics has been first used by Dekhtyar, et al.[11].

We discuss related work in more detail in Section 6.

DEFINITION 3. *Let* V *be a set of random variables. A* ***probabilistic interpretation (p-interpretation)*** *over* V *is a function* $I_V : dom(V) \rightarrow [0,1]$*, such that* $\sum_{\bar{x} \in dom(V)} I_V(\bar{x}) = 1$.

Given a set of random variables, a *p-interpretation* over it is any valid *point* probability distribution. An interval probability distribution function (**pdf**) $P : dom(V) \rightarrow$ C[0,1] represents a set of possible point probability distributions (a.k.a., *p-interpretations*). This corresponds to de Campos, et al.'s *instance*[8].

In the rest of the paper we adopt the following notation. Given a probability distribution $P : dom(V) \rightarrow$ C[0,1], for each $\bar{x} \in dom(V)$ we write $P(\bar{x}) = [l_{\bar{x}}, u_{\bar{x}}]$. Whenever $dom(V)$ is enumerated as $dom(V) = \{\bar{x}_1, \ldots \bar{x}_m\}$, we write $P(\bar{x}_i) = [l_i, u_i]$, $1 \leq i \leq m$. P is *complete* if it enumerates all possible events.

DEFINITION 4. *Let* V *be a set of random variables and* $P : dom(V) \rightarrow$ *C[0,1] a complete interval probability distribution function over* V*. A* p-interpretation I_V *satisfies* P *(*$I_V \models P$*) iff* $(\forall \bar{x} \in dom(V))(l_{\bar{x}} \leq I_V(\bar{x}) \leq u_{\bar{x}})$.

Let V *be a set of random variables and* $P' : X \rightarrow$ *C[0,1] an incomplete interval probability distribution function over* $X \subset dom(V)$*. A* probabilistic interpretation I_V *satisfies* P' *(*$I_V \models P'$*) iff* $(\forall \bar{x} \in X)(l_{\bar{x}} \leq I_V(\bar{x}) \leq u_{\bar{x}})$.

EXAMPLE 1. Consider a random variable v with domain $\{a, b, c\}$. Let probability distribution functions P_1, P_2 and P_3 and p-interpretations I_1, I_2, I_3 and I_4 be defined in the following table. We have $I_1 \models P_1$ and $I_1 \models P_2$; $I_2 \models P_2$ but $I_2 \not\models P_1$; $I_3 \models P_1$ but $I_3 \not\models P_2$ and, finally, $I_4 \not\models P_1$ and $I_4 \not\models P_2$. Also, $I_1 \not\models P_3$, $I_2 \not\models P_3$, $I_3 \not\models P_3$ and $I_4 \not\models P_3$.

P_1	P_2	P_3	I_1	I_2	I_3	I_4
$P_1(a) = [0.2, 0.3]$	$P_2(a) = [0.3, 0.6]$	$P_3(a) = [0.4, 0.5]$	$I_1(a) = 0.3$	$I_2(a) = 0.5$	$I_3(a) = 0.25$	$I_4(a) = 0.7$
$P_1(b) = [0.3, 0.45]$	$P_2(b) = [0.3, 0.4]$	$P_3(b) = [0.4, 0.5]$	$I_1(b) = 0.3$	$I_2(b) = 0.4$	$I_3(b) = 0.45$	$I_4(b) = 0.3$
$P_1(c) = [0.3, 0.7]$		$P_3(c) = [0.4, 0.5]$	$I_1(c) = 0.4$	$I_2(c) = 0.1$	$I_3(c) = 0.3$	$I_4(c) = 0$

DEFINITION 5. *An interval probability distribution function $P : dom(V) \rightarrow$ C[0,1] is **consistent** iff there exists a p-interpretation I_V, such that $I_V \models P$.*

Consider the interval probability distribution functions P_1, P_2 and P_3 described in Example 1. As we saw, $I_1 \models P_1$ and $I_1 \models P_2$, and thus, both P_1 and P_2 are consistent.

On the other hand, notice that any p-interpretation I satisfying P_3 must have $I(a) \geq 0.4$, $I(b) \geq 0.4$ and $I(c) \geq 0.4$, hence $I(a) + I(b) + I(c) \geq 1.2$, which contradicts the constraint $I(a) + I(b) + I(c) = 1$ on p-interpretations. Therefore, no p-interpretation would satisfy P_3 and thus, P_3 is *inconsistent*. The following theorem specifies the necessary and sufficient conditions for an interval probability distribution function to be consistent. Proofs for this and following theorems can be found in a technical report[27].

THEOREM 1. *Let V be a set of random variables and $P : dom(V) \rightarrow$ C[0,1] be a* complete *interval probability distribution function over V. Let $dom(V) = \{\bar{x}_1, \ldots, \bar{x}_m\}$ and $P(\bar{x}_i) = [l_i, u_i]$. P is **consistent** iff the following two conditions hold: (1) $\sum_{i=1}^{m} l_i \leq 1$; (2) $\sum_{i=1}^{m} u_i \geq 1$.*

Let $P' : X \rightarrow$ C[0,1] be an incomplete *interval probability distribution function over V. Let $X = \{\bar{x}_1, \ldots, \bar{x}_m\}$ and $P'(\bar{x}_i) = [l_i, u_i]$. P' is **consistent** iff $\sum_{i=1}^{m} l_i \leq 1$.*

DEFINITION 6. *Let $P : X \rightarrow$ C[0,1] be an interval probability distribution function over a set of random variables V. Let $X = \{\bar{x}_1, \ldots, \bar{x}_m\}$ and $P(\bar{x}_i) = [l_i, u_i]$. A number $\alpha \in [l_i, u_i]$ is **reachable** by P at $\bar{x}_i$ iff there exists a p-interpretation $I_V \models P$, such that $I(\bar{x}_i) = \alpha$.*

Reachability is another important property of interval pdfs. Intuitively points *unreachable* by an interval probability distribution function represent "dead weight"; they do not provide any additional information about possible satisfying p-interpretations. We note one important property of reachability: if two points α and β s.t. $l_i \leq \alpha < \beta \leq u_i$, are reachable by P at some point $\bar{x}$, then so are all point $\gamma \in [\alpha, \beta]$ [9].

DEFINITION 7. *Let $P : X \rightarrow$ C[0,1] be an interval probability distribution over a set V of random variables. P is called **tight** iff $(\forall \bar{x} \in X)(\forall \alpha \in [l_{\bar{x}}, u_{\bar{x}}])$ α is reachable by P at $\bar{x}$.*

Consider the interval pdf P_1 from Example 1. Although $P(c) = [0.3, 0.7]$, **no** p-interpretation $I \models P_1$ can have $I(c) = 0.7$, because knowing that $I(a) \geq 0.2$ and $I(b) \geq 0.3$ would lead to $I(a) + I(b) + I(c) \geq 1.3$ in violation of Definition 3. We conclude that P_1 is not tight.

In their work, de Campos, et al.[8] call tight intervals "reachable". As in their approach, we replace interval probability distributions that are not tight with their *tight equivalents*. This is done using a *tightening* operator.

DEFINITION 8. *Given an interval probability distribution P, an interval probability distribution P' is its* ***tight equivalent*** *iff (i) P' is tight and (ii) for each p-interpretation I, $I \models P$ iff $I \models P'$.*

A ***tightening*** *operator $\mathcal{T}$ takes as an input a consistent interval probability function $P : X \to$ C[0,1] and returns its tight equivalent $P' : X \to$ C[0,1].*

PROPOSITION 1. *Each complete interval probability distribution P has a unique* tight equivalent.

The key feature of the tightening operator is that it produces a new interval pdf that has **the same set of satisfying p-interpretations** as the input distribution function. We can compute the results of tightening efficiently as the following theorem shows.

THEOREM 2. *Let $P : dom(V) \to$ C[0,1] be a complete interval probability distribution function over a set of random variables V. Let $dom(V) = \{\bar{x}_1, \ldots, \bar{x}_m\}$ and $P(\bar{x}_i) = [l_i, u_i]$. Then* $(\forall 1 \leq i \leq m)$

$$\mathcal{T}(P)(\bar{x}_i) = [\max(l_i, 1 - \sum_{j=1}^{m} u_j + u_i), \min(u_i, 1 - \sum_{j=1}^{m} l_j + l_i)].$$

In the rest of the paper we assume that all ESPOs under consideration have *consistent* and *tight* probability distribution functions. The tightening operator allows us to replace any probability distribution function that is not tight with its tight equivalent. An ESPO $S = \langle T^+, V, P, C^+, \omega \rangle$ is called consistent iff P is consistent. Also, S is called tight iff P is tight.

5 EXTENDED PROBABILISTIC SEMISTRUCTURED ALGEBRA

Extended Probabilistic Semistructured Algebra (ESP-Algebra) is the query algebra for the ESPO model. It includes five major operations on probabilistic objects: selection, projection, Cartesian product, join and conditionalization. In our early paper[25], we defined these query algebra operations for pure interval probability distributions (without context and conditionals) in a generic way. Here, we ground the operations described there in the ESPO data model. The first four operations are extensions of standard relational algebra operations. However, these operations are expanded significantly in comparison with both classical relational algebra[20] and the definitions from Dekhtyar, et al.[10]. The conditionalization operation is specific to probabilistic databases and represents the procedure of constructing an ESPO containing a conditional probability distribution given an ESPO for some joint probability distribution. Introduced as a database operation by Dey and Sarkar[13] for a relational model with point probabilities, this operation had been extended to non-1NF databases by Dekhtyar, et al.[10] and considered for interval probabilities[25].

In the sections below, we describe each algebra operation. We base our examples on the elections in Sunny Hill that we have described in Section 2. Figure 3 shows different ESPOs representing a variety of polling data from Figures 1 and 2 and

ω: S_1
gender: men
party: Donkey
date: October 26

mayor	park	legalization	l	u
Donkey	yes	yes	0.44	0.52
Donkey	yes	no	0.12	0.16
Donkey	no	yes	0.08	0.12
Donkey	no	no	0.04	0.08
Elephant	yes	yes	0.05	0.1
Elephant	yes	no	0.01	0.02
Elephant	no	yes	0.03	0.04
Elephant	no	no	0.06	0.08
Rhino	yes	yes	0.02	0.04
Rhino	yes	no	0.01	0.03
Rhino	no	yes	0.03	0.05
Rhino	no	no	0.01	0.04

senate: Donkey

ω: S_2
date: October 23
gender: male
respondents: 238, {senate}
respondents: 195, {legalization}
overlap: 184

senate	legalization	l	u
Rhino	yes	0.04	0.11
Rhino	no	0.1	0.15
Donkey	yes	0.22	0.27
Donkey	no	0.09	0.16
Elephant	yes	0.05	0.13
Elephant	no	0.21	0.26

mayor: Donkey{senate, legalization}

ω S_3
locality: Sunny Hill
date: October 26

park	legalization	l	u
yes	yes	0.56	0.62
yes	no	0.14	0.2
no	yes	0.21	0.25
no	no	0.03	0.07

mayor: Donkey

ω: S_4
locality: South Side
date: October 12
sample: 323

mayor	l	u
Donkey	0.2	0.26
Elephant	0.42	0.49
Rhino	0.25	0.33

ω: S_5
locality: Downtown
date: October 12
sample: 275

mayor	l	u
Donkey	0.48	0.55
Elephant	0.25	0.3
Rhino	0.2	0.24

ω: S_6
locality: West End
date: October 12
sample: 249

mayor	l	u
Donkey	0.38	0.42
Elephant	0.34	0.4
Rhino	0.15	0.2

ω: S_7
locality: Sunny Hills
date: October 26
sample: 249

mayor	l	u
Donkey	0.33	0.39
Elephant	0.32	0.37
Rhino	0.25	0.3

Fig. 3. Sunny Hill pre-election polls in ESPO format.

more. We assume that all these objects have been inserted into the database in their current form, hence, each received a unique path id.

In a relational data model, a relation is defined as a collection of data tuples over the same set of attributes. In our model, an *Extended Semistructured Probabilistic relation* (ESP-relation) is a set of ESPOs and an *Extended Semistructured Probabilistic database* (ESP-database) is a set of ESP-relations. Grouping ESPOs into relations is done not based on structures, as is the case in the relational databases; ESPOs with different structures can co-exist in the same ESP-relation. In the examples below we consider ESP-relation $\mathcal{S} = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7\}$ consisting of ESPOs from Figure 3.

5.1 Selection

For each individual part of an ESPO we define a selection operation, namely: selection based on context, random variables, conditionals, probabilities and probability table. The first three types of selections, described in Section 5.1, when applied to an ESP-relation produce a subset of that relation. Individual ESPOs do not change (except for their paths). On the other hand, selections on probabilities or on probability tables (described in Section 5.1) may may return only parts of the probability tables. Table 1 lists some examples of queries that should be expressible as selection queries on ESPOs. For each question we describe the desired output of the selection operation.

Selection on Context, Random Variables and Conditionals In this section, we define the selection operations that do not alter the content of the selected objects. We start by defining the acceptable languages for selection conditions for these types of selects.

Recall that the universe $\mathcal{R}$ of context attributes consists of a finite set of attributes $A_1, \ldots A_n$ with domains $dom(A_1), \ldots, dom(A_n)$. With each attribute $A \in \mathcal{R}$ we associate a set $Pr(A)$ of allowed predicates. We assume that equality and inequality are allowed for all $A \in \mathcal{R}$. The definitions below formalize the selection operations on a single ESPO.

Table 1. Selection queries to ESPOs.

#	Query	Answer
1.	"What information is available about voter attitudes on October 26?"	Set of ESPOs that have date: October 26 in their context.
2.	"What are other voting intentions of people who choose to vote Donkey for mayor?"	Set of ESPOs which have as a conditional mayor=Donkey.
3.	"What information is known about voter intentions in the mayoral race?"	Set of ESPOs that contain mayor in the set of participating random variables
4.	"What voting patterns are likely to occur with probability between 0.2 and 0.3?"	In the probability table of each ESPO, the rows with probability values guaranteed to be between 0.2 and 0.3 are found. If such rows exist, they form the probability table of the ESPO that is returned by the query.
5.	"With what probability are voters likely to choose a Donkey mayor and Elephant Senator?	Set of all ESPOs that contain mayor and senate random variables, with the probability tables of each containing only the rows where mayor=Donkey and senate=Elephant.
6.	"Find all distributions based on more than 200 responses about senate vote."	Set of ESPOs that contain senate random variable and responses = X with $X > 200$ is associated with it in the context.
7.	"How do people who intend to vote Donkey for mayor plan to vote for the park construction ballot initiative?"	Set of ESPOs that contain park random variable and conditional mayor=Donkey is associated with it.

DEFINITION 9. *An **atomic context selection condition** is an expression c of the form "$A\ \ Q\ \ x\ \ (Q(A,x))$", where $A \in \mathcal{R}$, $x \in dom(A)$ and $Q \in Pr(A)$. An **atomic participation selection condition** is an expression c of the form "$v \in V$", where $v \in \mathcal{V}$ is a random variable. An **atomic conditional selection condition** is one of the following expressions: "$u = \{x_1, \ldots x_h\}$" or "$u \ni x$" where $u \in \mathcal{V}$ is a random variable and $x, x_1, \ldots, x_h \in dom(u)$. We slightly abuse notation and write "$u = x$" instead of "$u = \{x\}$". An **extended atomic context selection condition** is an expression c/W^* where c is an* atomic context selection condition *and $W^* \subseteq \mathcal{V}$ is a set of random variables. An **extended atomic conditional selection condition** is an expression c/W^* where c is an* atomic conditional selection condition *and $W^* \subseteq \mathcal{V}$ is a set of random variables.*

DEFINITION 10. *Let $S = \langle T^+, V, P, C^+, \omega \rangle$ and $S' = \langle T^+, V, P, C^+, \omega' \rangle$ be two ESPOs with $\omega' =$ "$\sigma_c(\omega)$".*

*Let $c = Q(A,x)$ be an atomic context selection condition, then $\sigma_c(S) = \{S'\}$ **iff** there exists a tuple $(A, a, W) \in T^+$, such that $(a,x) \in Q$; otherwise $\sigma_c(S) = \emptyset$.*

*Let $c : v \in V$ be an atomic participation selection condition, then $\sigma_c(S) = \{S'\}$ **iff** $v \in V$; otherwise $\sigma_c(S) = \emptyset$.*

*Let $c : u = \{x_1, \ldots, x_h\}$ be an atomic conditional selection condition, then $\sigma_c(S) = \{S'\}$ **iff** $(u, X, W) \in C^+$ and $X = \{x_1, \ldots, x_h\}$; otherwise $\sigma_c(S) = \emptyset$.*

*Let $c : u \ni x$ be an atomic conditional selection condition, then $\sigma_c(S) = \{S'\}$ **iff** $(u, X, W) \in C^+$ and $x \in X$; otherwise $\sigma_c(S) = \emptyset$.*

*Let $c = Q(A, x)/W^*$ be an extended atomic context selection condition, then $\sigma_c(S) = \{S'\}$ **iff** there exists a tuple $(A, a, W) \in T^+$ such that (i) $(a, x) \in Q$; (ii) $W^* \subseteq W$; otherwise $\sigma_c(S) = \emptyset$.*

*Let $c : u = \{x_1, \ldots, x_h\}/W^*$ be an extended atomic conditional selection condition, then $\sigma_c(S) = \{S'\}$ **iff** $(u, X, W) \in C^+$, $X = \{x_1, \ldots, x_h\}$, and $W^* \subseteq W$; otherwise $\sigma_c(S) = \emptyset$.*

*Let $c : u \ni x/W^*$ be an extended atomic conditional selection condition, then $\sigma_c(S) = \{S'\}$ **iff** $(u, X, W) \in C^+$, $x \in X$ and $W^* \subseteq W$; otherwise $\sigma_c(S) = \emptyset$.*

We note that, whenever an ESPO satisfies any of the selection conditions described above, its context, participating variables, probability table and conditional are returned intact. The only part of the ESPO that changes is its path, reflecting the application of the selection operation to the object. The semantics of atomic selection conditions discussed so far can be extended to Boolean combinations in a straightforward manner: $\sigma_{C \wedge C'}(S) = \sigma_C(\sigma_{C'}(S))$ and $\sigma_{C \vee C'}(S) = \sigma_C(S) \vee \sigma_{C'}(S)$. Finally, for an ESP-relation $\mathcal{S}$, $\sigma_C(\mathcal{S}) = \bigcup_{S \in \mathcal{S}}(\sigma_C(S))$.

EXAMPLE 2. Consider our ESP-relation $\mathcal{S}$ (Figure 3). Below are some possible queries to this relation and their results (we specify the unique ids of the ESPOs that match the query).

id	Type	Query	Result
Q1	context	$\sigma_{\mathsf{date}=\mathsf{October26}}(\mathcal{S})$	$\{S_1, S_3, S_7\}$
Q2	participation	$\sigma_{\mathsf{mayor} \in V}(\mathcal{S})$	$\{S_1, S_4, S_5, S_6, S_7\}$
Q3	conditionals	$\sigma_{\mathsf{senate}=\{\mathsf{Donkey}\}}(\mathcal{S})$	$\{S_1\}$
Q4	ext. context	$\sigma_{\mathsf{respondents}>200/\{\mathsf{senate}\}}(\mathcal{S})$	$\{S_2\}$
Q5	ext. context	$\sigma_{\mathsf{gender}=\mathsf{men}/\{\mathsf{mayor,park}\}}(\mathcal{S})$	$\{S_1\}$
Q6	ext. conditional	$\sigma_{\mathsf{mayor}=\{\mathsf{Donkey}\}/\{\mathsf{senate}\}}(\mathcal{S})$	$\{S_2\}$
Q7	ext. conditional	$\sigma_{\mathsf{mayor}=\{\mathsf{Donkey}\}/\{\mathsf{senate,house}\}}(\mathcal{S})$	$\emptyset$

Selection on Probabilities and Probability Tables The two types of selections introduced in this section are more complex. The result of a selection operation of either type depends on the content of the probability table, which itself is considered as a relation (each row being a single record). In the process of performing the probabilistic selection or selection on the probability table (see questions 4 and 5, Table 1, respectively), each row of the probability table is examined individually to determine whether it satisfies the selection condition. A row is retained in the answer if it does, otherwise it is thrown out. Thus, such selection operations may yield ESPOs with *incomplete* probability tables. As the selection condition relates only to the content of the probability table of an ESPO, its context, participating random variables, and conditionals are preserved.

DEFINITION 11. *An **atomic probabilistic table selection condition** is an expression of the form $v = x$ where $v \in \mathcal{V}$ and $x \in dom(v)$. An **atomic probabilistic selection condition** is an expression of one of the two forms: (i) l op α; (ii) u op α; where $\alpha \in [0, 1]$ and $\mathsf{op} \in \{=, \neq, \leq, \geq, <, >\}$.*

DEFINITION 12. *Let $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO, $V = \{v_1, \ldots, v_k\}$, and let $c : v = x$ be an atomic probabilistic table selection condition.*

If $v \in V$, then (assuming $v = v_i, 1 \leq i \leq k$) the result of selection from S on c, $\sigma_c(S)$ is a semistructured probabilistic object $S' = \langle T^+, V, P', C^+, \omega'\rangle$, where $\omega' =$ "$\sigma_c(\omega)$" and

$$P'(y_1, \ldots, \mathbf{y_i}, \ldots, y_k) = \begin{cases} P(y_1, \ldots, \mathbf{y_i}, \ldots, y_k) & \textit{if } \mathbf{y_i} = x; \\ \textbf{\textit{undefined}} & \textit{if } \mathbf{y_i} \neq x. \end{cases}$$

Consider the ESPO S_1 from Figure 3. The leftmost ESPO of Figure 4 shows the result of the selection query on probability table: $\sigma_{\mathsf{park=yes}}(S_1)$ ("find the probability of all voting outcomes where respondents support the park ballot initiative"). The result of this query is computed as follows: the context, list of conditionals and participating random variables remain the same, while the probability table now contains only the rows that satisfy the selection condition and the path changes to reflect the selection operation. If the same query is applied to the entire relation $\mathcal{S}$, the result contains two ESPOs constructed from S_1 and S_3: only these ESPOs have participating random variable park (and rows for park=yes).

ω: $\sigma_{\mathsf{path=yes}}(S_1)$
gender: men
party: Donkey
date: October 26

mayor	park	legalization	l	u
Donkey	yes	yes	0.44	0.52
Donkey	yes	no	0.12	0.16
Elephant	yes	yes	0.05	0.1
Elephant	yes	no	0.01	0.02
Rhino	yes	yes	0.02	0.04
Rhino	yes	no	0.01	0.03

senate: Donkey

ω: $\sigma_{u>0.14}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
respondents: 195, {legalization}
overlap: 184

senate	legalization	l	u
Rhino	no	0.1	0.15
Donkey	yes	0.22	0.27
Donkey	no	0.09	0.16
Elephant	no	0.21	0.26

mayor: Donkey{senate, legalization}

ω: $\sigma_{l<0.11}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
respondents: 195, {legalization}
overlap: 184

senate	legalization	l	u
Rhino	yes	0.04	0.11
Rhino	no	0.1	0.15
Donkey	no	0.09	0.16
Elephant	yes	0.05	0.13

mayor: Donkey{senate, legalization}

Fig. 4. Selection on probability table and probabilities.

DEFINITION 13. *Let $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO, and $c : l \text{ op } \alpha$ $(c : u \text{ op } \alpha)$ a probabilistic atomic selection condition. Let $\bar{x} \in dom(V)$. The result of selection from S on c is defined as follows: $\sigma_{P \text{ op } \alpha}(S) = S' = \langle T^+, V, P', C^+, \omega'\rangle$, where $\omega' =$ "$\sigma_c(\omega)$" and*

$$P'(\bar{x}) = \begin{cases} P(\bar{x}) & \textit{if } l_{\bar{x}} \text{ op } \alpha \ (u_{\bar{x}} \text{ op } \alpha); \\ \textbf{\textit{undefined}} & \textit{otherwise.} \end{cases}$$

The center and the rightmost ESPOs on Figure 4 represent the results of selections on probabilities: $\sigma_{u>0.14}(S_2)$ and $\sigma_{l<0.11}(S_2)$ respectively. In both cases, the results of the selection keep the same context, conditionals and participating random variables, while the probability table is modified to retain only the rows where the upper (lower) bound on the probability interval satisfies the selection condition. The result of $\sigma_{u>0.14}(\mathcal{S})$ would contain seven ESPOs: every object in $\mathcal{S}$ contains rows where upper bound on probability is greater that 0.14. The result of $\sigma_{l<0.11}(\mathcal{S})$

contains two ESPOs constructed from S_1 and S_2: only these SPOs had rows with lower probability less than 0.11.

Different selection operations (described in this section and in Section 5.1) commute.

THEOREM 3. *Let c and c' be two selection conditions and let $\mathcal{S}$ be a semistructured probabilistic relation. Then $\sigma_c(\sigma_{c'}(\mathcal{S})) = \sigma_{c'}(\sigma_c(\mathcal{S}))$.*

5.2 Projection

Projection in classical relational algebra removes columns from the relation and, if needed, collapses duplicate tuples. ESPOs consist of four different components that can be affected by projection operation. We distinguish between three different types of projection here: on context, on conditionals and on participating random variables, the latter affecting probability table as well.

There are two issues that need to be addressed when defining projection on context. First, contexts may contain numerous copies of relational attributes. Hence, projecting out a particular attribute from the context of an ESPO should result in *all* copies if this attribute being projected out. The second issue is the fact that in extended context, different attributes are associated with different participating random variables. To address these two issues we define two types of projection on context. The first operation is similar to standard relational projection, while the second operation works by removing associations between context attributes and random variables.

DEFINITION 14. *Let $F = \{A_1, \ldots, A_k\}$ be a set of* context attributes *and $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO.* ***Projection of** S **on** F, denoted $\pi_F(S)$ is an ESPO $S' = \langle T^{+'}, V, P, C^+, \omega'\rangle$, where $T^{+'} = \{(A, a, W) | (A, a, W) \in T^+,\ A \in F\}$ and $\omega' =$ "$\pi_F(\omega)$".*

DEFINITION 15. *Let $F^+ = \{(A_1, W_1) \ldots, (A_k, W_k)\}$ be a set of pairs where for $1 \le i \le k$, A_i is a context attribute and $W_i \subseteq \mathcal{V}$. Let $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO.* ***Projection of** S **on** F^+, denoted $\pi_{F^+}(S)$, is an ESPO $S' = \langle T^{+'}, V, P, C^+, \omega'\rangle$, where $T^{+'} = \{(A, a, W') | (A, a, W) \in T^+,\ A = A_i \in \{A_1, \ldots A_k\}$* for some $1 \le i \le k$, and $\emptyset \ne W' = W \cap W_i\}$ *and $\omega' =$ "$\pi_{F^+}(\omega)$".*

Given an ESPO S and a set of pairs F^+ as described in Definition 15, the projection operation proceeds as follows. The set of context attributes *to keep* which comes from F^+ specifies for each attribute the list of random variables for which it is allowed to be kept. The projection operation (i) removes from the input ESPO S all attributes not in F^+ and (ii) for each instance $(A, a, W) \in T^+$ of attribute A_i s.t. $(A_i, W_i) \in F^+$, it removes all references in W that are not in W_i. If $W \cap W_i = \emptyset$, then (A, a, W) is omitted from the projection. Projection operations on conditionals can be defined similarly.

DEFINITION 16. *Let $U = \{u_1, \ldots, u_k\} \subseteq \mathcal{V}$ be a set of* random variables *and $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO.* ***Projection of** S **on** U, denoted $\pi_{C:U}(S)$, is an ESPO $S' = \langle T^+, V, P, C^{+'}, \omega'\rangle$, where $C^{+'} = \{(u, X, W) | (u, X, W) \in C^+,$* and $u \in U\}$ *and $\omega' =$ "$\pi_{C:U}(\omega)$".*

DEFINITION 17. *Let $U^+ = \{(u_1, W_1) \ldots, (u_k, W_k)\}$ be a set of pairs where for all $1 \leq i \leq k$, $u_i \in \mathcal{V}$ and $W_i \subseteq \mathcal{V}$. Let $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO. **Projection of** S **on** U^+, denotes $\pi_{C:U^+}(S)$, is an ESPO $S' = \langle T^+, V, P, C^{+'}, \omega'\rangle$, where $C^{+'} = \{(u, X, W')|(u, X, W) \in T^+,\ u = u_i \in \{u_1, \ldots, u_k\}$, and $\emptyset \neq W' = W \cap W_i\}$; $\omega' =$ "$\pi_{C:U}(\omega)$".*

Symbol "C" is used in the notation to distinguish the projection operation from the projection on the set of participating random variables, to be defined below.

Let us now define the most intricate projection operation, projection on the set of random variables. When defining this operation, we need to keep in mind the following: (i) projection is only allowed if at least one random variable remains in the resulting set of participating random variables, (ii) projecting out a random variable v should result in removal of v from the extended context and conditionals, (iii) projecting out a random variable v should remove this variable from the probability table, i.e., the underlying probability distribution function changes.

DEFINITION 18. *Let $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO, and let $V^* \subset \mathcal{V}$. **Projection of** S **on** V^*, denoted $\pi_{V^*}(S)$, is defined as follows:*
(1) $V^* \cap V = \emptyset$: $\pi_{V^*}(S) = \emptyset$.
(2) $V^* \cap V = V' \neq \emptyset$: $\pi_{V^*}(S) = S' = \langle T^{+'}, V', P', C^{+'}, \omega'\rangle$, where
- $T^{+'} = \{(A, a, W')|(A, a, W) \in T^+$ and $\emptyset \neq W' = W \cap V^*\}$;
- $C^{+'} = \{(u, X, W')|(u, X, W) \in C^+$ and $\emptyset \neq W' = W \cap V^*\}$;
- $P' : dom(V') \rightarrow$ ***C[0,1]***. *For all $\bar{x'} \in dom(V')$ and $(\bar{x'}, \bar{x''}) \in dom(V)$,*
$P'(\bar{x'}) = [\min_{I \models P}(\sum_{(\bar{x'},\bar{x''}) \in dom(V)} I(\bar{x'}, \bar{x''}))\ , \max_{I \models P}(\sum_{(\bar{x'},\bar{x''}) \in dom(V)} I(\bar{x'}, \bar{x''}))]$; *and*
- $\omega' =$ "$\pi_{V^*}(\omega)$".

This definition requires a careful explanation. Let $S = \langle T^+, V, P, C^+, \omega\rangle$ be an ESPO, and let $V^* \subseteq \mathcal{V}$ be the set of projection random variables. The computation of $\pi_{V^*}(S)$ proceeds as follows. First, we check if the intersection of V, the set of participating random variables of S, and V^* is empty. If it is, we return the empty set as the answer. If $V' = V \cap V^*$ is not empty, we build the projection as follows.
(i) The new set of participating random variables is V'.
(ii) The new context $T^{+'}$ and conditionals $C^{+'}$ are produced from T^+ and C^+ respectively, by eliminating all random variables *not from* V' from the extensions (associations). Context entries (conditionals) from T^+ (C^+) associated only with variables *not from* V' are eliminated from $T^{+'}$ ($C^{+'}$).
(iii) Finally, the new probability table function is defined as follows. The function must range over $dom(V')$. Since $V' \subseteq V$, associated with each value $\bar{x'} \in dom(V')$, is a set of values $(\bar{x'}, \bar{x''}) \in dom(V)$, where $\bar{x''}$ ranges over $dom(V - V')$. Given a p-interpretation $I \models P$, for each $\bar{x'} \in dom(V')$ we compute the probability assigned to it by P as $I(\bar{x'}) = \sum_{\bar{x''} \in dom(V - V')} I(\bar{x'}, \bar{x''})$. We know that the probability of $\bar{x'}$ has to range between the minimal and maximal value of $I(\bar{x'})$, for all $I \models P$. This interval, $[\min_{I \models P} I(\bar{x'}), \max_{I \models P} I(\bar{x'})]$, is the value of the new probability distribution function P' on $\bar{x'}$.

While the computation of the new set of participating random variables, context, and conditionals according to Definition 18 is straightforward, computing the new probability table requires solving a number of optimization problems (finding $\min$s and $\max$s of $\sum I(\bar{x'}, \bar{x''})$ for all $\bar{x'}$), which seems like a fairly tedious task. However, it turns out that these optimization problems have analytical solutions.

THEOREM 4. *Let* $S = \langle T^+, V, P, C^+, \omega\rangle$ *be an ESPO and* $V^* \subseteq \mathcal{V}$. *Let* $V \cap V^* \neq \emptyset$ *and* $S' = \langle T^{+'}, V', P', C^{+'}, \omega'\rangle = \pi_{V^*}(S)$. *Let* $P''(x') = [\sum_{(\bar{x'},\bar{x''})\in dom(V)} l_{(\bar{x'},\bar{x''})}, \min(1, \sum_{(\bar{x'},\bar{x''})\in dom(V)} u_{(\bar{x'},\bar{x''})})]$. *Then* $P' = \mathcal{T}(P'')$.

EXAMPLE 3. Figure 5 illustrates the computation of projection $\pi_{\{\mathsf{senate}\}}(S_2)$ on participating random variables. The first step is the removal of all other random variables from the probability table. Next, the duplicate rows of the new probability table are collapsed and the probability intervals are added. After that, tightening is performed to find the true intervals. We then exclude respondents:195 from the context as it is not associated with variable senate and disassociate legalization with conditionals.

ω: S_2
date: October 23
gender: male
respondents: 238, {senate}
respondents: 195, {legalization}
overlap: 184

senate	legalization	l	u
Rhino	yes	0.04	0.11
Rhino	no	0.1	0.15
Donkey	yes	0.22	0.27
Donkey	no	0.09	0.16
Elephant	yes	0.05	0.13
Elephant	no	0.21	0.26

mayor: Donkey{senate, legalization}

$\Longrightarrow$

ω: $\pi_{\{\mathsf{senate}\}}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
respondents: 195, {legalization}
overlap: 184

senate	l	u
Rhino	0.04	0.11
Rhino	0.1	0.15
Donkey	0.22	0.27
Donkey	0.09	0.16
Elephant	0.05	0.13
Elephant	0.21	0.26

mayor: Donkey{senate, legalization}

$\Longrightarrow$

ω: $\pi_{\{\mathsf{senate}\}}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
overlap: 184

senate	l	u
Rhino	0.14	0.26
Donkey	0.31	0.43
Elephant	0.26	0.39

mayor: Donkey{senate}

$\Longrightarrow$

ω: $\pi_{\{\mathsf{senate}\}}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
overlap: 184

senate	l	u
Rhino	0.18	0.26
Donkey	0.35	0.43
Elephant	0.31	0.39

mayor: Donkey{senate}

Fig. 5. Projection on the participating random variables.

5.3 Conditionalization

Conditionalization was introduced into relational algebra for a probabilistic data model by Dey and Sarkar[13]. It is the operation of computing a conditional probability distribution, given a joint probability distribution. To simplify the definition below, we employ the following notation. Let $V = \{v_1, \ldots, v_n\}$ be a set of random variables and let $v \in V$ and $V' = V - \{v\}$. Let $I : dom(V) \to [0, 1]$

be a p-interpretation. Let $X = \{x_1, \ldots x_m\} \subset dom(v)$ and $\bar{y} \in dom(V')$. Let $I_X(\bar{y}) = \sum_{i=1}^{m} I(\bar{y}, x_i)$.

DEFINITION 19. *Let $S = \langle T^+, V, P, C^+, \omega \rangle$ be an ESPO, $|V| > 1$, $v \in V$ and $c : v = \{x_1, \ldots, x_m\}$ be a conditional selection condition. Then, the result of* ***conditionalization of*** *S* ***on*** *c, denoted $\mu_c(S)$, is the ESPO $S' = \langle T^+, V', P', C^{+'}, \omega' \rangle$, where*

- *$V' = V - \{v\}$. Without loss of generality, we assume further that $V = \{v_1, \ldots, v_n\}$, $v = v_n$ and therefore $V' = \{v_1, \ldots, v_{n-1}\}$.*
- *$C^{+'} = C^+ \cup \{(v, X, V')\}$, where $X = \{x_1, \ldots, x_m\}$.*
- *$P' : dom(V') \to$ C[0,1] is defined as*

$$P'(\bar{y}) = \left[\min_{I \models P} \left(\frac{I_X(\bar{y})}{\sum_{y' \in dom(V')} I_X(\bar{y'})} \right), \max_{I \models P} \left(\frac{I_X(\bar{y})}{\sum_{y' \in dom(V')} I_X(\bar{y'})} \right) \right].$$

- *$\omega' = \mu_c(\omega)$.*

Given a p-interpretation I, $\frac{I_X(\bar{y})}{\sum_{y' \in dom(V')} I_X(\bar{y'})}$ is the conditional probability of $\bar{y}$ given $v = v_n \in X$. From the definition above, it follows that in order to compute the result of conditionalization of an ESPO (in particular, in order to compute the resulting probability distribution) a number of non-linear optimization problems have to be solved. As it turns out, the new probability distribution can be computed directly (i.e., both minimization and maximization problems that need to be solved have analytical solutions)[8,9].

THEOREM 5. *Let $S = \langle T^+, V, P, C^+, \omega \rangle$ be an ESPO, $c : v = \{x_1, \ldots, x_m\}$ be a conditional selection condition and $v \in V$. Let $V' = V - \{v\}$, $X = \{x_1, \ldots, x_m\}$ and $\bar{y} \in dom(V')$. The result of the conditionalization is denoted $S' = \mu_c(S) = \langle T^+, V', P', C^{+'}, \omega' \rangle$. If we define $l[X]_{\bar{y}}$ and $u[X]_{\bar{y}}$ as follows:*

$$l[X]_{\bar{y}} = \max \left(\sum_{x \in X} l_{(\bar{y},x)} \; ; \; 1 - \sum_{\bar{y'} \neq \bar{y} \text{ or } x' \notin X} u_{(\bar{y'},x')} \right),$$

$$u[X]_{\bar{y}} = \min \left(1 - \sum_{\bar{y'} \neq \bar{y} \text{ or } x' \notin X} l_{(\bar{y'},x')} \; ; \; \sum_{x \in X} u_{(\bar{y},x)} \right),$$

then the following expression correctly computes the lower and upper bounds of the conditional probability distribution for the resulting ESPO object. Note that the theorem assumes that denominators of the expressions in the formula are non-zero. This is the case if any of the lower bounds are non-zero.

$$P'(\bar{y}) = \left[\frac{l[X]_{\bar{y}}}{\min\left(1 - \sum_{x' \notin X} l_{(\bar{y'},x')} \, , \, \sum_{\bar{y^*} \neq \bar{y} \, , \, x \in X} u_{(\bar{y^*},x)} + l[X]_{\bar{y}}\right)} \right.,$$

$$\frac{u[X]_{\bar{y}}}{\max\left(\sum_{\bar{y^*}\neq\bar{y}\ ,\ x\in X} l_{(\bar{y^*},x)} + u[X]_{\bar{y}}\ ,\ 1-\sum_{x'\notin X} u_{(\bar{y'},x')}\right)}\Bigg].$$

The proof of this theorem can be found in early paper[9]. The following example illustrates how the conditionalization operation works.

EXAMPLE 4. Consider the ESPO S_2 in Figure 3. In this example, we show the computation of the conditionalization $\mu_{\{\mathsf{legalization=yes}\}}(S_2)$, as shown in Figure 6. First we collapse all the rows that do not satisfy the condition $\mathsf{legalization} = \mathsf{yes}$ into one row. Next, we do a tightening operation on the new probability distribution. Then we normalize, which means that we must find the minimum and maximum values of the expressions of the form $\frac{\mathsf{l(w,yes)}}{\mathsf{l(Rhino,yes)+l(Donkey,yes)+l(Elephant,yes)}}$ for $w \in \{\mathsf{Rhino}, \mathsf{Donkey}, \mathsf{Elephant}\}$ over all p-interpretations $I \models P$.

Let us determine the lower bound for $x = \mathsf{Rhino}$. Consider the following function f of three variables: $f(x, y, z) = \frac{x}{x+y+z}$. (In this paper, we discard the case when $x + y + z = 0$.) For positive x, y and z, we could rewrite the function as $f(x, y, z) = \frac{1}{1+\frac{y+z}{x}}$. So, in order to minimize f we need to minimize x and maximize y+z. In this case, we need to minimize $\mathsf{l(Rhino, yes)}$ and maximize $\mathsf{l(Donkey, yes)}+ \mathsf{l(Elephant, yes)}$, i.e., $\mathsf{l(Rhino, yes)} = 0.04$ and $\mathsf{l(Donkey, yes)}+ \mathsf{l(Elephant, yes)} = \min(0.27 + 0.13, 1 - 0.04 - 0.49) = 0.4$. Then the minimum value of $\frac{\mathsf{l(Rhino,yes)}}{\mathsf{l(Rhino,yes)+l(Donkey,yes)+l(Elephant,yes)}}$ is $\frac{0.04}{0.04+0.4} = 0.09$.

ω: S_2
date: October 23
gender: male
respondents: 238, {senate}
respondents: 195, {legalization}
overlap: 184

senate	legalization	l	u
Rhino	yes	0.04	0.11
Rhino	no	0.1	0.15
Donkey	yes	0.22	0.27
Donkey	no	0.09	0.16
Elephant	yes	0.05	0.13
Elephant	no	0.21	0.26

mayor: Donkey{senate, legalization}

$\Longrightarrow$

ω: $\mu_{\{\mathsf{legalization=yes}\}}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
overlap: 184

senate	legalization	l	u
Rhino	yes	0.04	0.11
Donkey	yes	0.22	0.27
Elephant	yes	0.05	0.13
/	no	0.49	0.57

mayor: Donkey{senate, legalization}

$\Longrightarrow$

ω: $\mu_{\{\mathsf{legalization=yes}\}}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
overlap: 184

senate	l	u
Rhino	0.09	0.36
Donkey	0.48	0.53
Elephant	0.12	0.30

mayor: Donkey{senate}
legalization: yes{senate}

$\Longrightarrow$

ω: $\mu_{\{\mathsf{legalization=yes}\}}(S_2)$
date: October 23
gender: male
respondents: 238, {senate}
overlap: 184

senate	l	u
Rhino	0.17	0.36
Donkey	0.48	0.53
Elephant	0.12	0.30

mayor: Donkey{senate}
legalization: yes{senate}

Fig. 6. Conditionalization operation.

Similarly, we can determine the upper bound for x = Rhino. We need to maximize I(Rhino, yes) and minimize I(Donkey, yes) + I(Elephant, yes), i.e., I(Rhino, yes) $= 0.11$ and I(Donkey, yes) + I(Elephant, yes) $= \max(0.22 + 0.05, 1 - 0.11 - 0.57) = 0.32$. Then the maximum value of $\frac{\mathsf{I(Rhino,yes)}}{\mathsf{I(Rhino,yes)+I(Donkey,yes)+I(Elephant,yes)}}$ is $\frac{0.11}{0.11+0.32} = 0.36$. We can apply similar operations for x = Donkey and x = Elephant. After that, the tightening operation is performed. Finally, we exclude respondents:195 from the context as it is associated with legalization variable and add legalization=yes to the conditionals. The resulting ESPO is shown in the bottom right of Figure 6.

We note, however, that Jaffray [16] has shown that conditionalizing interval probabilities is a tricky matter: the set of point probability distributions represented by P' contains distributions that do not correspond to any distribution in P. This appears to be an inescapable feature of conditionalization of interval pdfs in the possible worlds semantics. However, the result of conditionalization still gives the tightest possible probability intervals for each instance, and is therefore useful in practice. Thus, conditionalization is included in ESP-algebra with a caveat to the users to view its results with caution.

5.4 Cartesian Product and Join

The Cartesian product of two ESPOs can be viewed as the joint probability distribution of the random variables from both objects. As only point probabilities were used [10], an assumption of independence was made between the random variables in the SPOs being combined. Probability distribution functions considered here are interval, so this restriction is removed.

Probabilistic conjunctions are interval functions (operations) that are used to compute the probability of a conjunction of two events given the probabilities of individual events. Typically, each probabilistic conjunction operation would have an *underlying assumption* about the relationship between the events involved, such as *independence*, *ignorance*, *positive* or *negative correlation*. Probabilistic conjunctions ($\otimes_\alpha$)[a] were introduced by Lakshmanan, et al. [19], and used in their Cartesian product operation. Our definitions are borrowed from Dekhtyar, et al. [11] and Lakshmanan, et al. [19].

Cartesian Product Since different probabilistic conjunction operations compute the probabilities of conjunction of two events in different ways, there is no unique Cartesian product operation. Rather, for each probabilistic conjunction $\otimes_\alpha$ we define a Cartesian product operation $\times_\alpha$.

DEFINITION 20. *Let* $S = \langle T^+, V, P, C^+, \omega \rangle$ *and* $S' = \langle T^{+'}, V', P', C^{+'}, \omega \rangle$ *be two ESPOs. Let* $V = \{v_1, \ldots, v_n\}$, $V' = \{v'_1, \ldots, v'_m\}$, $U = \{u \in \mathcal{V} | (u, X, W) \in C^+\}$, $U' = \{u' \in \mathcal{V} | (u', X', W') \in C^{+'}\}$. *$S$ and S' are **Cartesian product-compatible** iff (i)* $V \cap V' = \emptyset$; *(ii)* $U \cap V' = \emptyset$, *and (iii)* $V \cap U' = \emptyset$.

We require that the sets of participating random variables be disjoint. (The other case is handled by the join operation.) We also want the set of random variables

[a] For example, $([l_1, u_1] \otimes_{ig} [l_2, u_2]) = [\max(0, l_1 + l_2 - 1), \min(u_1, u_2)]$ for ignorance relationship, and $([l_1, u_1] \otimes_{ind} [l_2, u_2]) = [l_1 \cdot l_2, u_1 \cdot u_2]$ for independence relationship.

found in the conditionals of one ESPO to be disjoint from the participating variables of the other. For example, Cartesian product of the probability distribution of mayor votes for respondents who will vote Donkey for senate with the probability distribution of senate votes is not allowed.

DEFINITION 21. *Let* $S = \langle T^+, V, P, C^+, \omega \rangle$ *and* $S' = \langle T^{+'}, V', P', C^{+'}, \omega' \rangle$ *be two Cartesian-product compatible ESPOs. Let* $V = \{v_1, \ldots, v_n\}$, $V' = \{v'_1, \ldots, v'_m\}$, $U = \{u \in \mathcal{V} | (u, X, W) \in C^+\}$, $U' = \{u' \in \mathcal{V} | (u', X', W') \in C^{+'}\}$. *Let* $\otimes_\alpha$ *be some probabilistic conjunction. The* Cartesian product *of S and* S'*, denoted* $S \times_\alpha S'$*, is defined as* $S \times_\alpha S' = S'' = \langle T^{+''}, V'', P'', C^{+''}, \omega'' \rangle$*, where*

- $V'' = V \cup V'$;
- $T^{+''} = \{(A, a, W) | [(A, a, W) \in T^+$ and <u>no</u> $(A, a, W') \in T^{+'}]$ **or** $[(A, a, W) \in T^{+'}$ and <u>no</u> $(A, a, W') \in T^+]$ **or** $[(A, a, W_1) \in T^+$ and $(A, a, W_2) \in T^{+'}$ and $W = W_1 \cup W_2]\}$;
- $P'' : dom(V'') \rightarrow$ *C[0,1] is defined as follows. Let* $\bar{x} \in dom(V)$, $\bar{x'} \in dom(V')$ *(hence* $(\bar{x}, \bar{x'}) \in dom(V'')$*). Then* $P''((\bar{x}, \bar{x'})) = P(\bar{x}) \otimes_\alpha P'(\bar{x'})$;
- $C^{+''} = \{(u, X, W) | [(u, X, W) \in C^+$ and <u>no</u> $(u, X, W') \in T^{+'}]$ **or** $[(u, X, W) \in T^{+'}$ and <u>no</u> $(u, X, W') \in T^+]$ **or** $[(u, X, W_1) \in T^+$ and $(u, X, W_2) \in T^{+'}$ and $W = W_1 \cup W_2]\}$*; and*
- $\omega'' =$ "$\omega \times_\alpha \omega'$".

In Cartesian product the contexts and the conditionals of the two input ESPOs are united; If a particular context attribute or a conditional appears in both ESPOs, then their association lists are merged. The new set of participating variables is the union of the two original sets. Finally, the probability interval for each instance (row) of the new probability table is computed by applying the probabilistic conjunction operation to the appropriate rows of the two original tables.

Join Similar to Cartesian product, join in ESP-Algebra computes the joint probability distribution of the input ESPOs. The difference is that join is applicable to the ESPOs that have *common participating random variables*. Let $S = \langle T^+, V, P, C^+, \omega \rangle$ and $S' = \langle T^{+'}, V', P', C^{+'}, \omega \rangle$, and let $V_c = V \cap V' \neq \emptyset$ and participating random variables of S are not conditioned in S' and vice versa. If these conditions are satisfied, we call S and S' *join-compatible*.

DEFINITION 22. *Let* $S = \langle T^+, V, P, C^+, \omega \rangle$ *and* $S' = \langle T^{+'}, V', P', C^{+'}, \omega' \rangle$ *be two ESPOs. Let* $V = \{v_1, \ldots, v_n\}$, $V' = \{v'_1, \ldots, v'_m\}$, $U = \{u \in \mathcal{V} | (u, X, W) \in C^+\}$, $U' = \{u' \in \mathcal{V} | (u', X', W') \in C^{+'}\}$. S *and* S' *are* ***join-compatible*** *iff (i)*$V \cap V' = V_c \neq \emptyset$*; (ii)* $U \cap V' = \emptyset$*, and (iii)* $V \cap U' = \emptyset$.

Consider three vectors $\bar{x} \in dom(V - V_c)$, $\bar{y} \in dom(V_c)$, and $\bar{z} \in dom(V' - V_c)$. The join of S and S' is the joint probability distribution $P''(\bar{x}, \bar{y}, \bar{z})$ of V and V', or, more specifically, of $V - V_c$, V_c and $V' - V_c$. To construct this joint distribution, we recall from probability theory that under assumption α about the relationship between the random variables in V and V' and independence between variables in $V - V_c$ and in $V' - V_c$, we have $p(\bar{x}, \bar{y}, \bar{z}) = p(\bar{x}, \bar{y}) \otimes_\alpha p(\bar{z}|\bar{y})$ and, symmetrically, $p(\bar{x}, \bar{y}, \bar{z}) = p(\bar{x}|\bar{y}) \otimes_\alpha p(\bar{y}, \bar{z})$. $p(\bar{x}, \bar{y})$ is stored in P, the probability table of S. $p(\bar{z}|\bar{y})$ is the conditional probability that can be found by conditioning $p(\bar{y}, \bar{z})$ (stored in P') on $\bar{y}$. The second equality can be exploited in the same manner.

This gives rise to two families of join operations, left join ($\ltimes_\alpha$) and right join ($\rtimes_\alpha$), defined as follows.

DEFINITION 23. *Let* $S = \langle T^+, V, P, C^+, \omega\rangle$ *and* $S' = \langle T^{+'}, V', P', C^{+'}, \omega'\rangle$ *be two join-compatible ESPOs. Let* $V_c = V \cap V' \neq \emptyset$. *We define the operations of left join of* S *and* S', *denoted* $S \ltimes_\alpha S'$, *and right join of* S *and* S', *denoted* $S \rtimes_\alpha S'$, *under the assumption* α *as follows:* $S \ltimes_\alpha S' = S'' = \langle T^{+''}, V'', P'', C^{+''}, \omega''\rangle$;
$S \rtimes_\alpha S' = S''' = \langle T^{+''}, V'', P''', C^{+''}, \omega'''\rangle$, *where*

- $V'' = V \cup V'$;
- $T^{+''} = \{(A, a, W) | [(A, a, W) \in T^+$ and $\underline{\text{no}}$ $(A, a, W') \in T^{+'}]$ **or** $[(A, a, W) \in T^{+'}$ and $\underline{\text{no}}$ $(A, a, W') \in T^+]$ **or** $[(A, a, W_1) \in T^+$ and $(A, a, W_2) \in T^{+'}$ and $W = W_1 \cup W_2]\}$;
- $P'', P''' : dom(V'') \longrightarrow$ C[0,1]. *For all* $\bar{w} \in dom(V'')$; $\bar{w} = (\bar{x}, \bar{y}, \bar{z})$; $\bar{x} \in dom(V - V_c), \bar{y} \in dom(V_c)$, $\bar{z} \in dom(V' - V_c)$: *let* $S_{\bar{y}} = \mu_{V_c = \bar{y}}(S) = \langle T^+, V - V_c, P_{\bar{y}}, C^+_{\bar{y}}\rangle$ *and* $S'_{\bar{y}} = \mu_{V_c=\bar{y}}(S') = \langle T^{+'}, V' - V_c, P'_{\bar{y}}, C^{+'}_{\bar{y}}\rangle$. $P''(\bar{w}) = P_{\bar{y}}(\bar{x}) \otimes_\alpha P'(\bar{y}, \bar{z})$; $P'''(\bar{w}) = P((\bar{x}, \bar{y})) \otimes_\alpha P'_{\bar{y}}(\bar{z})$.
- $C^{+''} = \{(u, X, W) | [(u, X, W) \in C^+$ and $\underline{\text{no}}$ $(u, X, W') \in T^{+'}]$ **or** $[(u, X, W) \in T^{+'}$ and $\underline{\text{no}}$ $(u, X, W') \in T^+]$ **or** $[(u, X, W_1) \in T^+$ and $(u, X, W_2) \in T^{+'}$ and $W = W_1 \cup W_2]\}$; *and*
- $\omega'' = \omega \ltimes_\alpha \omega'$; $\omega''' = \omega \rtimes_\alpha \omega'$.

EXAMPLE 5. Consider the two ESPOs S_2 and S_3 in Figure 3. They are joint probability distributions for (senate, legalization) and (park, legalization), respectively. However, in some circumstances we may want to combine these two ESPOs and obtain the joint probability distribution for all three random variables. We may apply a join operation to these ESPOs since they are join-compatible according the definition. We show the left join under the assumption of independence, $S_2 \ltimes_{ind} S_3$, as follows.

The join operation combines three operations: conditionalization, selection and Cartesian product. First, we need to calculate the results for conditionalization of the left operand (i.e. S_2) on the set of common variables (in this case, legalization). Second, we do selections on probability table for all the possible values of the common variables, namely, $\sigma_{legalization=yes}(S_3)$ and $\sigma_{legalization=no}(S_3)$. Third, Cartesian product operations on corresponding ESPOs [b] are applied based on the values of the common variables, namely, $\mu_{legalization=yes}(S_2) \times_\alpha \sigma_{legalization=yes}(S_3)$ and $\mu_{legalization=no}(S_2) \times_\alpha \sigma_{legalization=no}(S_3)$. We assume that the random variables in these two ESPOs are independent when we apply probability conjunctions. Finally, we union the resulting ESPOs and apply a tightening operation. The final result is shown in the right side of the figure.

6 RELATED WORK

This work builds on the work of many people in two fields: imprecise probabilities and probabilistic databases. The overlap between these two fields is still small, so we address them separately. Databases that handle imprecise probabilities are surveyed in Section 6.2.

[b] Prior to applying Cartesian product operation, we project legalization = yes and legalization = no out of the conditionals.

ω:	S_2		
date:	October 23		
gender:	male		
senate	legalization	l	u
Rhino	yes	0.04	0.11
Rhino	no	0.1	0.15
Donkey	yes	0.22	0.27
Donkey	no	0.09	0.16
Elephant	yes	0.05	0.13
Elephant	no	0.21	0.26
mayor:	Donkey		

ω	S_3		
locality:	Sunny Hill		
date:	October 26		
park	legalization	l	u
yes	yes	0.56	0.62
yes	no	0.14	0.2
no	yes	0.21	0.25
no	no	0.03	0.07
major:	Donkey		

$\Longrightarrow$

ω: $S_2 \ltimes S_3$				
locality:	Sunny Hill			
date:	October 26			
gender:	male			
senate	park	legalization	l	u
Rhino	yes	yes	0.09	0.22
Rhino	yes	no	0.03	0.06
Rhino	no	yes	0.04	0.09
Rhino	no	no	0.01	0.02
Donkey	yes	yes	0.27	0.33
Donkey	yes	no	0.03	0.07
Donkey	no	yes	0.11	0.13
Donkey	no	no	0.01	0.02
Elephant	yes	yes	0.07	0.19
Elephant	yes	no	0.06	0.11
Elephant	no	yes	0.03	0.07
Elephant	no	no	0.01	0.04
major:	Donkey			

Fig. 7. Join operation (left join) in ESP-Algebra

6.1 Interval Probabilities

Imprecise probabilities have attracted the attention of researchers for quite a while now, as documented by the Imprecise Probability Project[22]. Walley's seminal work[21] makes the case for interval probabilities as the means of representing uncertainty. In his book, Walley talks about the computation of conditional probabilities of events. His semantics is quite different from ours, as Walley constructs his theory of imprecise probabilities based on gambles and betting, expressed as lower and upper previsions on the sets of events. Conditional probabilities are also specified via gambles by means of *conditional previsions*. A similar approach to Walley's is found in the work of Biazzo, Gilio, et al.[2,3] where they extend the theory of imprecise probabilities to incorporate logical inference and default reasoning.

Walley[21] calls consistency and tightness properties "avoiding sure loss", and "coherence", respectively. Biazzo and Gilio[2] also use the term "g-coherence" as a synonym for "avoiding sure loss". Their work focuses on checking g-coherence and propagation of lower/upper conditional probabilities *without assuming that the conditioning events have positive probability.* The terminology that we have adopted originated in the work of Dekhtyar, Ross and Subrahmanian on a specialized semantics for probability distributions used in their Temporal Probabilistic Database model[11]. However, the semantics presented here is a significant generalization of their semantics. The possible worlds semantics for interval probabilities also occurs in Givan, Leach and Dean's discussion of Bounded Parameter Markov Decision Processes[14].

de Campos, Huete and Moral[8] study probability intervals as a tool to represent uncertain information. They introduce similar definitions of consistency and tightness, which they call reachability. They develop a calculus for probability intervals, including combination, marginalization and conditioning. They also explore the relationship of their formalism with other theories of uncertainness, such as lower and upper probabilities. When they define their conditioning operation, however, they switch back and apply lower and upper probabilities to uncertain information instead of probability intervals, and give a definition of a conditioning operation only for bidimensional probability intervals. Ours follows their definition.

A more direct approach to introducing interval probabilities is found in the work of Weichselberger[23] who extends the Kolmogorov axioms of probability theory to the case of interval probabilities. Building on Kolmogorov probability theory, the interval probability semantics is defined for a σ-algebra of random events. Weichselberger defines two types of interval probability distributions over this σ-algebra: R-Probabilities, similar to our consistent interval pdfs, and F-Probabilities, similar to our tight interval pdfs. In his semantics an event is specified as a Boolean combination of atomic events from some set Ω. Each event partitions the set of possible worlds into two sets: those in which the event has occurred and those in which it has not. A lower bound on the probability that an event has occurred is immediately an upper bound on the probability that it has not occurred. Thus, for F-probabilities, Weichselberger's analogues of our tight p-interpretations, lower bounds uniquely determine upper bounds.

Weichselberger completes his theory with two definitions of conditional probability: "intuitive" and "formal". His "intuitive" definition semantically matches our Definition 19. On the other hand, the "formal" definition specifies the probability interval for $P(A|B)$ as $[\frac{\min(P(AB))}{\min P(B)}, \frac{\max(P(AB))}{\max P(B)}]$, which is somewhat different from our Theorem 5. There, to determine the lower bound we minimize the numerator and *maximize* the denominator. Similarly, for the upper bound, we maximize the numerator and *minimize* the denominator.

In our semantics, atomic events have the form "random variable X_1 takes value a_1 and random variable X_2 takes value a_2 and ... and random variable X_m takes value a_m." The negation of such an event is the disjunction of all other atomic events that complete the joint probability distribution of random variables $X_1, \dots, X_m$. Our interval pdfs specify only the probability intervals for such atomic events, without explicitly propagating them onto the negations. This means that even for tight interval pdfs, both upper and lower bounds are necessary in all but marginal cases, as illustrated in Figure 8.

X	l	u
a	0.3	0.4
b	0.4	0.5
c	0.2	0.3

X	l	u
a	0.3	0.35
b	0.4	0.45
c	0.2	0.27

Fig. 8. Lower bounds do not uniquely define upper bounds for tight interval pdfs.

Interval probability distributions of discrete random variables generate a set of linear constraints on the acceptable probability values for individual instances. This set of linear constraints, however, is quite simple. It consists of constraints specifying that the probabilities of individual instances must fall between the given lower and upper bounds and a constraint that specifies that the sum of all probabilities must be equal to 1. It is possible, however, to study more complex collections of constraints on possible worlds. Significant work in this area has been done by Cano and Moral[6].

A further and much more comprehensive survey of interpretations can be found in the Imprecise Probabilities Project[18]. None of the work surveyed there, however, discusses database management issues.

6.2 Probabilistic Databases

Cavallo and Pittarelli[7] were among the first to address the problem of storing and querying probabilistic information in a database. Their probabilistic relations resemble a single ESPO probability table. Their data model requires that the probabilities for all the tuples in a relation add up to exactly 1. As a result, unlike ours, their model requires a separate relation for each object. Barbara, Garcia-Molina and Porter[1] propose a new approach to managing probabilistic information. In their model, certain attributes in a relation can be designated as *stochastic* and (possibly joint) probability distributions can be associated with these attributes. The analogue of their non-stochastic attributes in our framework is context, while stochastic attributes are represented as participating random variables. The model of Barbara, et al., was relational, and so the probability distributions stored in a single probabilistic relation had to be of the same structure.

Dey, et al.[13] introduced a 1NF probabilistic relational model and relational algebra. A tuple in their model is analogous to a single row of a probability table in ours, and their probabilistic relation can contain multiple probability distributions. Both Barbara, et al.[1] and Dey, et al.[13] use point probabilities and assume that all events/random variables in their models are independent. Lakshmanan, et al. introduce ProbView[19], a probabilistic database management system. In ProbView, probability distributions are interval, and the assumption of independence of events is replaced with the introduction of probabilistic conjunctions (and disjunctions), implementing different assumptions about the relationships between the events. Based on the ProbView model, Dekhtyar, Ross and Subrahmanian develop Probabilistic Temporal Databases (TP-Databases)[11], a special-purpose probabilistic database model for managing temporal uncertainty. In this work, a semantics of interval probability distributions similar to the one used in ESPO model is introduced, and the concept of *tightness* appears for the first time in the database literature.

Dey, et al.[13] first introduce the conditionalization operation in a probabilistic database model. Dekhtyar, Goldsmith and Hawkes also use this operation in their Semistructured Probabilistic Algebra[10]. In both works, conditionalization is performed on *point probability distributions* of discrete random variables; The operation itself is fairly straightforward for point probability. The conditionalization operation as a database operation for probability intervals was not included in data models until recently by Goldsmith, Dekhtyar and Zhao[15]. However, Jaffray [16] has shown that interval conditional probability estimates are not perfect, and that the unfortunate consequence of this is that conditionalizing is not commutative: $P((A|B)|C) \neq P(A|(B|C))$ for many A, B, and C. Thus, a conditionalization operation is included in ESP-Algebra with the caveat that users must take care in the use and interpretation of the result.

Two approaches to semistructured probabilistic data management are closely related to ours: the ProTDB[24] and the PIXml[17] frameworks. In ProTDB[24], Nierman and Jagadish extend the XML data model by associating a probability with each element by modifying regular non-probabilistic DTDs. They provide two ways of modifying non-probabilistic DTDs, either by introducing to every element a probability attribute Prob to specify the probability of the particular element existing at the specific location of the XML document or by attaching a new sub-element called Dist to each element. One of the drawbacks is that in their model probabilities are always conditional. All other probabilities are assumed to be independent. Hung, et al.[17] proposed a probabilistic interval XML data model with two types of semantics for uncertain data. The global interpretation is a distribution over an entire XML document, while the local interpretation specifies an object probability function for each non-leaf object. They also propose a path expression-based query language to access stored information. This approach overcomes some drawbacks presented in the ProTDB[24]. The major difference between it and our work is that the PIXml[17] is concerned with representation of uncertainty in the structure of XML documents. At the same time, the ESPO model provides a semistructured data type for storing probability distributions found in different applications. Hung, et al., use our conditionalization formulae for their computations of conditional probabilities. This makes the two approaches comparable: Our ESPO objects can be represented as their probabilistic XML. We can also represent their probabilistic XML documents as joint probability distributions, and thus embed them into ESPO model. While ESPOs are representable in XML, our definitions of the model and ESP-Algebra do not rely on a specific representation.

7 CONCLUSIONS AND FUTURE WORK

The Extended Semistructured Probabilistic Objects and Extended Semistructured Probabilistic Algebra introduced here represent a flexible database framework for storing and managing diverse probabilistic information. While such operations as probabilistic table selection, projection and conditionalization have been defined via the underlying semantics (i.e., in terms of satisfying p-interpretations), we have been able to provide direct ways of computing the results of these operations in each case, which lead to clear and efficient algorithms.

We have designed and implemented a semistructured probabilistic database management system (SPDBMS) on top of a RDBMS, and reported a performance evaluation of SPDBMS for each query algebra operation[26]. Currently we are working on implementing a query optimizer for the SPDBMS. Implementation of an extension to the SPDBMS to handle probability intervals has been underway. We are also studying data fusion and conflict resolution problems that arise in this framework.

8 Acknowledgement

This work was partially supported by NSF grant CCR-0100040 and ITR-0219924. This paper is a significant extension of our previous work[9,15,25]. We thank the anonymous reviewers of our papers, whose suggestions improved this paper.

References

1. D. Barbara, H. Garcia-Molina and D. Porter. (1992) The Management of Probabilistic Data, *IEEE Trans. on Knowledge and Data Engineering*, Vol. 4, pp. 487–502.
2. V. Biazzo and A. Gilio. (2000) A generalization of the fundamental theorem of de Finetti for imprecise conditional probability assessments, *International Journal of Approximate Reasoning*, 24(2), pp. 251–272.
3. V. Biazzo, A. Gilio, T. Lukasiewicz and G. Sanfilippo. (2001) Probabilistic Logic under Coherence, Model-Theoretic Probabilistic Logic, and Default Reasoning, *Proc. ECSQARU'2001, LNAI*, Vol. 2143, pp. 290–302.
4. G. Boole. (1854) *The Laws of Thought*, Macmillan, London.
5. T. Bray, J. Paoli and C.M. Spreberg-McQueen. (Eds.) (1998) Extensible Markup Language (XML) 1.0, *World Wide Web Consortium Recommendation*, http://www.w3.org/TR/1998/REC-xml-19980210.
6. A. Cano and S. Moral. (2000) Using probability trees to compute marginals with imprecise probabilities, *Universidad de Granada, Escuela Técnica Superior de Ingenieria Informática technical report, DECSAI-00-02-14*.
7. R. Cavallo and M. Pittarelli. (1987) The Theory of Probabilistic Databases, *Proc. VLDB'87*, pp. 71-81.
8. L. M. de Campos, J. F. Huete and S. Moral. (1994) Probability Intervals: A Tool for Uncertain Reasoning, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 2(2), pp. 167–196, 1994.
9. A. Dekhtyar and J. Goldsmith. (2002) Conditionalization for Interval Probabilities, *Proc. Workshop on Conditionals, Information, and Inference*, May, 2002.
10. A. Dekhtyar, J. Goldsmith and S.R. Hawkes. (2001) Semistructured Probabilistic Databases, in *Proc. SSDBM'2001*.
11. A. Dekhtyar, R. Ross and V.S. Subrahmanian. (2001) Temporal Probabilistic Databases, I: Algebra, *ACM Transactions on Database Systems*, vol 26, 1, pp. 41–95.
12. A. Dekhtyar and V.S. Subrahmanian. (2000) Hybrid Probabilistic Logic Programs, *Journal of Logic Programming*, vol 43, 3, pp. 187–250.
13. D. Dey and S. Sarkar. (1996) A Probabilistic Relational Model and Algebra, *ACM Transactions on Database Systems*, Vol. 21, 3, pp. 339–369.
14. R. Givan, S. Leach and T. Dean. (2000) Bounded-Parameter Markov Decision Processes, *Artificial Intelligence*, Vol. 122, 1-2, pp. 71–109.
15. J. Goldsmith, A. Dekhtyar and W. Zhao. (2003) Can Probabilistic Databases Help Elect Qualified Officials?, *Proc. Florida AI Research Symposium,* pp. 501–505.
16. J.Y. Jaffray. (1992) Bayesian Updating and Belief Functions. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(5), pp. 1144–1152.
17. E. Hung, L. Getoor and V.S. Subrahmanian. (2003) Probabilistic Interval XML, *Proc. International Conference on Database Theory*. pp. 361–377.
18. H. Kyburg. (1998) Interval-valued probabilities. G. de Cooman, P. Walley, and F. G. Cozman, editors, *Imprecise Probabilities Project*.
19. V.S. Lakshmanan, N. Leone, R. Ross and V.S. Subrahmanian. (1997) ProbView: A Flexible Probabilistic Database System. *ACM Transactions on Database Systems*, Vol. 22, No. 3, pp.419–469.
20. R. Ramakrishnan and J Gehrke. (1999) *Database Management Systems*, 2nd Ed. McGraw-Hill.

21. P. Walley. (1991). Statistical Reasoning with Imprecise Probabilities. Chapman and Hall, 1991.
22. G. de Cooman and P. Walley. *The Imprecise Probabilistic Project* URL: http://ippserv.rug.ac.be
23. K. Weichselberger. (1999) The theory of interval-probability as a unifying concept for uncertainty. *Proc. 1st International Symp. on Imprecise Probabilities and Their Applications*.
24. A. Nierman and H. V. Jagadish. (2002) ProTDB: Probabilistic Data in XML. *Proc. the 28th International VLDB Conference*. Hong Kong, China.
25. W. Zhao, A. Dekhtyar and J. Goldsmith. (2003) Query Algebra for Interval Probabilities. *Proc. 14th International Conference on Database and Expert Systems Applications*, 527–536.
26. W. Zhao, A. Dekhtyar and J. Goldsmith. (2003) A Framework for Management of Semistructured Probabilistic Data. Department of Computer Science, University of Kentucky. *Tech Report TR385-03*, http://www.cs.uky.edu/∼wzhao0/papers/TR385-03.pdf.
27. W. Zhao, A. Dekhtyar and J. Goldsmith. (2003) Databases for Interval Probabilities. Department of Computer Science, University of Kentucky. *Tech Report TR386-03*, http://www.cs.uky.edu/∼wzhao0/papers/TR386-03.pdf.