Query Evaluation in Probabilistic Relational Databases

Esteban Zimányi

Université Libre de Bruxelles, 50 Av. F. Roosevelt, C.P. 175/02, 1050 Brussels, Belgium, e-mail: ezimanyi@ulb.ac.be

Abstract

This paper describes a generalization of the relational model in order to capture and manipulate a type of probabilistic information. Probabilistic databases are formalized by means of logic theories based on a probabilistic first-order language proposed by Halpern. A sound a complete method is described for evaluating queries in probabilistic theories. The generalization proposed can be incorporated into existing relational systems with the addition of a component for manipulating propositional formulas.

1 Introduction

The introduction of *incomplete* and *uncertain information* in relational databases has been an active area of research (see e.g. [43] for a survey).

The first attempts to introduce incomplete information were the study of *null values* (e.g., [3,11,30,7]) and *disjunctive information* [38,12,18,19,42] . The definition of closure assumptions in the presence of disjunctive information (e.g., [21,32,27]) has also led to the field of disjunctive logic programming. Minker [22] surveys the developments in this field.

Representing and handling uncertain information have also been active areas of research in the last two decades. Theories for handling uncertain information include probabilistic approaches, Shafer's Evidence Theory, Zadeh's Possibility Theory, Cohen's Theory of Endorsements, in addition to all the work done in non-monotonic logics. To review these theories' basic concepts, we refer for example to [36] .

In the context of relational databases, research has focused on uncertainty under two different approaches. The first one uses Zadeh's fuzzy sets and possibility theory to define *fuzzy databases*. The second one follows a probabilistic framework to define *probabilistic databases*.

Fuzzy databases were proposed as an attempt to extend the classical relational model for manipulating imprecise data values such as "John's salary is around 60,000" or "John has a high salary". Fuzzy set theory and fuzzy logic (e.g., [39–41]) provide a mathematical framework to deal with such extended data values. Important work has been done on the study of relational databases in the light of fuzzy set theory including areas such as generalizing classical relational operators, query language design, query evaluation, and integrity constraint modeling. For an entry point to this subject's bibliography, we refer for example to [28] .

For modeling uncertainty in relational databases, the probabilistic approach has been much less studied than the fuzzy approach. Probabilistic models for relational databases have been proposed in [6,2,1,26] , but there is still work to be done. In Section 6, we review related approaches concerning probabilistic extensions of deductive databases and logic programming.

As in the case of the fuzzy approach, two types of probabilistic information may be introduced in relational databases. The first one allows to represent attributes whose exact value is unknown but with a probability distribution; for example, "Ralph will teach a course which is either Calculus or Physics, the former with probability 0.8 and the latter with probability 0.2". The manipulation of this kind of information is studied in [1] . The second type of information allows to represent events whose probability lies in the interval $[0, 1]$; for example, "the probability that Paul takes Calculus is 0.8".

This paper describes an extension of the relational model in order to capture and manipulate the second type of probabilistic information. We define *probabilistic relations* as generalizations of classical relations with a supplementary attribute $w_R(\bar{t})$, indicating the probability that tuple $\bar{t}$ belongs to relation R.

Like classical relational databases are formalized with first-order logic theories [29,30,10] , we formalize probabilistic databases by means of probabilistic logic theories based on a probabilistic language proposed by Halpern [9] . Given a first-order language for reasoning about a domain and a formula ϕ of this logic, the probabilistic language allows formulas of the form $w(\phi) \geq \frac{1}{2}$ which can be interpreted as "the probability that ϕ is satisfied is greater than or equal to $\frac{1}{2}$". Once probabilistic databases are formalized with probablistic theories, a sound and complete method for query evaluation is proposed.

The remaining sections are as follows. In Section 2, we discuss, by means of examples, the representational aspects and the semantics of probabilistic relational databases. Section 3 gives an introduction to the formal preliminaries in probabilistic languages, the formalization of probabilistic databases with probabilistic theories, and the definition of queries in these theories. We introduce also in that section a running example is used throughout the paper. We give

then, in Sections 4 and 5, a sound and complete query evaluation algorithm for probabilistic theories. Finally, Section 6 discusses related works while Section 7 summarizes the results of the paper and indicates some directions for future research.

2 Probabilistic databases

Information of a stochastic nature is very common in real-world applications. Modeling probabilistic information is thus a significant aspect in database and artificial intelligence applications. To generalize the relational model with uncertain information, we must distinguish two types of uncertainties: uncertainties in *data values* and uncertainties in the *association between values*. An example of uncertainty in data values with a relation teaches(professor,course) is "John teaches a course which is Algebra with probability 0.8 and Calculus with probability 0.2". An example of uncertainty in the association between values with a relation takes(student,course) is "the probability that Peter takes the Databases course is 0.9". Uncertainties in data values and uncertainties in the association between values can also be combined.

We study in this paper the representation and manipulation of the second type of uncertainty. We define *probabilistic relations* as generalizations of classical relations whith a supplementary attribute $w_R(\bar{t})$, indicating the probability that tuple $\bar{t}$ belongs to relation R. An example is given in Figure 1. This

takes

student	*course*	w_R
Tom	Physics	1.0
Tom	Algebra	0.9
John	Physics	0.5
Anne	Algebra	0.6

Fig. 1. A probabilistic relation.

relation represents, for example, that Tom surely takes Physics, and that the probability that he takes Algebra is 0.9. Thus, the probability that he does not take Algebra is 0.1. Probabilistic relations are written in tabular form as in Figure 1, or in a set notation as

$$\begin{aligned} takes = \{&(\text{Tom,Physics})/1.0, (\text{Tom,Algebra})/0.9, \\ &(\text{John,Physics})/0.5, (\text{Anne,Algebra})/0.6\}. \end{aligned}$$

Semantics for probabilistic relations can be stated as follows. Consider relation *takes* of Figure 1, and suppose that a student takes a course independently of the courses taken by the other students. Suppose also that the relation

is interpreted under a closed world assumption, specifying that every pair (student,course) not present in the relation has probability 0.

Under these assumptions, relation *takes* represents $2^3 = 8$ possible situations with certain information, varying from the situation where only (Tom,Physics) belongs to the relation to the situation where the 4 tuples belong to the relation. Each of these "possible worlds" can be represented by a classical relation with an associated probability, computed as the product of the probabilities for the presence or absence of each tuple of relation *takes*. These possible worlds are given in Figure 2.

(Tom,Physics) (Tom,Algebra) (John,Physics) (Anne,Algebra)	(Tom,Physics) (Tom,Algebra) (John,Physics)	(Tom,Physics) (Tom,Algebra) (Anne,Algebra)	(Tom,Physics) (John,Physics) (Anne,Algebra)
0.27	0.18	0.27	0.03
(Tom,Physics) (Tom,Algebra)	(Tom,Physics) (John,Physics)	(Tom,Physics) (Anne,Algebra)	(Tom,Physics)
0.18	0.02	0.03	0.02

Fig. 2. Possible worlds of relation *takes*.

To formalize probabilistic databases we use a probabilistic language proposed by Halpern [9] , a two-sorted logic where a sort $\mathcal{O}$ describes objects of the domain and a sort $\mathcal{F}$ describes probabilities. Variables of sorts $\mathcal{O}$ and $\mathcal{F}$ are denoted, respectively, by x and by x^f.

We use probabilistic theories to formalize probabilistic databases. As in Reiter's relational theories [30] , each relation is associated with an object predicate of the same name, having as many places as there are attributes in the relation. Also, probabilistic theories contain a non empty set of simple types, modeling different domains for the variables, and a set of *extension axioms* associated to each object predicate. Relation *takes* of Figure 1 can be represented with the following extension axioms:

$$
\begin{aligned}
&(\forall x)(\forall y)(takes(x,y) \rightarrow \\
&\qquad (x = \text{Tom} \wedge y = \text{Physics}) \vee (x = \text{Tom} \wedge y = \text{Algebra}) \vee \\
&\qquad (x = \text{John} \wedge y = \text{Physics}) \vee (x = \text{Anne} \wedge y = \text{Algebra})), \qquad (1)\\
&(\forall x)(\forall y)((x = \text{Tom} \wedge y = \text{Physics}) \rightarrow takes(x,y)), \qquad (2)\\
&(\forall x)(\forall y)(\forall z^f)(w(takes(x,y)) = z^f \wedge 0 < z^f < 1 \leftrightarrow \\
&\qquad (x = \text{Tom} \wedge y = \text{Algebra} \wedge z^f = 0.9) \vee \\
&\qquad (x = \text{John} \wedge y = \text{Physics} \wedge z^f = 0.5) \vee \\
&\qquad (x = \text{Anne} \wedge y = \text{Algebra} \wedge z^f = 0.6)). \qquad (3)
\end{aligned}
$$

The first extension axiom realizes the closure of the relation by stating all the

tuples belonging to it. Thus it can be deduced, for example, that Anne does not take Physics. The second extension axiom states the tuples belonging surely to the relation, i.e., the tuples having probability 1.0. Finally, the third extension axiom specifies the tuples belonging to the relation with probability greater than 0 and less than 1.0. Although these extension axioms could be stated differently, the proposed notation facilitates the subsequent development.

Probabilistic theories contain another type of axioms dealing with the independence of probabilities. One such an axiom could be

$$w(takes(\text{Tom}, \text{Algebra}) \wedge takes(\text{John}, \text{Physics})) = \\ w(takes(\text{Tom}, \text{Algebra})) \times w(takes(\text{John}, \text{Physics})),$$

stating that the two events are independent. Another axiom could be

$$w(takes(\text{Tom}, \text{Algebra}) \leftrightarrow takes(\text{Anne}, \text{Algebra})) = 0.9$$

stating that, with 0.9 probability, Tom takes Algebra iff Anne also does. Finally, the next axiom

$$w(takes(\text{Tom}, \text{Algebra}) \mid teaches(\text{Peter}, \text{Algebra})) = 0.1$$

states that the probability that Tom takes Algebra given that Peter teaches Algebra is 0.1.

This paper only considers the case where all the facts in the database are independent. Relaxing this constraint is discussed in Sections 6 and 7.

3 Formal preliminaries

3.1 Probabilistic languages

We give now an introduction to Halpern's *probabilistic languages* [1] which will be used to formally define probabilistic databases in the next section. This section is largely inspired from [9] .

A probabilistic language $\mathcal{L}$ is a two-sorted language where a sort $\mathcal{O}$ describes objects of the domain and a sort $\mathcal{F}$ describes probabilities.

[1] Halpern defines three types of probabilistic languages; the languages used here are called type-2 languages.

Sort $\mathcal{O}$ contains finitely many constants a, b, c, ... , a countable family of variables x, y, ..., and no function symbols. Sort $\mathcal{F}$ contains three constants 0, 1, and -1, representing the corresponding real numbers, a countable family of variables x^f, y^f, ... , and two binary function symbols $+$ and $\times$, representing addition and multiplication. Constants and variables of sort $\mathcal{O}$ (resp. $\mathcal{F}$) are called *object* (resp. *field*) constants and variables.

Language $\mathcal{L}$ contains finitely many predicates of sort $\mathcal{O} \times \cdots \times \mathcal{O}$, called *object predicates*. These predicates include object equality, denoted by $=$, and a distinguished set of unary predicates called *simple types*; these simple types allow to model the domains of standard relational theory. There are also two predicates of sort $\mathcal{F} \times \mathcal{F}$, denoted by $>$ and by $=$, representing the predicates greater than and field equality.

In a probabilistic language $\mathcal{L}$, *object terms*, *field terms*, and *formulas* are defined inductively as follows. Object terms are object variables and constants. Field terms are formed by starting with field variables or constants and terms of the form $w(\varphi)$, where φ is an arbitrary formula, and closing off under field function application so that if t_1, t_2 are field terms, then $t_1 + t_2$ and $t_1 \times t_2$ are field terms. Formulas are formed as in many-sorted logics. We distinguish two types of formulas: *first-order formulas* are formulas without field terms, whereas *probabilistic formulas* are arbitrary formulas of $\mathcal{L}$.

The connectives $\vee$, $\rightarrow$, and $\exists$ are defined in terms of $\wedge$, $\neg$, and $\forall$ as usual. Similarly, $-$, $/$, $\sqrt{\ }$, $<$, $\geq$, $\leq$, and k (where k is an integer) are defined in terms of the basic elements of $\mathcal{F}$. In addition, *simple ground field terms* (denoted sgf-terms) are defined by induction as follows: we start with 0, 1, and -1, and then we close off so that if t_1 and t_2 are sgf-terms, then so are $t_1 + t_2$, $t_1 - t_2$, $t_1 \times t_2$, t_1/t_2 if $t_2 \neq 0$, and $\sqrt{t_1}$ if $t_1 \geq 0$.

The semantics of probabilistic languages is based on the concept of structures. A *structure* of a probabilistic language $\mathcal{L}$ is a tuple $M = (\mathcal{D}, S, \pi, \mu)$ where $\mathcal{D}$ is the domain, S is a set of *states* or *possible worlds*, for each state $s \in S$, $\pi(s)$ assigns to object constants and predicates, respectively, constants and relations of the right arity over $\mathcal{D}$, and μ is a discrete probability function assigning a probability to each possible world of S. For any $A \subseteq S$, we define $\mu(A) = \sum_{s \in A} \mu(s)$. As usual, a valuation v assigns to every variable x a constant $v(x)$ from $\mathcal{D}$.

Given a probability structure M, a state s, and a valuation v, we can associate with every object (resp. field) term t an element $[t]_{(M,s,v)}$ of $\mathcal{D}$ (resp. of $\mathbb{R}$) and with every formula φ a truth value, writing $(M, s, v) \models \varphi$ if the value *true* is associated with φ by (M, s, v). We just give a few clauses of the definition,

since they follow the lines of first-order logic:

– $(M, s, v) \models P(x)$ iff $v(x) \in \pi(s)(P)$;
– $(M, s, v) \models (t_1 = t_2)$ iff $[t_1]_{(M,s,v)} = [t_2]_{(M,s,v)}$;
– $(M, s, v) \models (\forall x)\varphi$ iff $(M, s, v[x/d]) \models \varphi$ for all $d \in \mathcal{D}$;
– $[w(\varphi)]_{(M,s,v)} = \mu(\{s' \in S \mid (M, s', v) \models \varphi\})$ for all $s \in S$.

We say $M \models \varphi$ if $(M, s, v) \models \varphi$ for all states s in M and all valuations v, and say φ is *valid*, and write $\models \varphi$, if $M \models \varphi$ for all structures M.

Halpern also gives an axiomatization of probabilistic languages. The axiom system is composed of several parts. First, it includes axioms and inference rules for first-order logic reasoning. Second, in order to reason about probabilities, which are real numbers, the axiom system contains all instances of a standard complete axiomatization for real closed fields (e.g. [35]). Finally, the axiom system includes the axioms for probabilistic reasoning as follows. If φ and ψ are arbitrary formulas, then

P1. $\varphi \rightarrow w(\varphi) = 1$, if every object predicate symbol of φ appears in an argument ψ of a probability term of the form $w(\psi)$.
P2. $w(\varphi) \geq 0$.
P3. $w(\varphi \wedge \psi) + w(\varphi \wedge \neg\psi) = w(\varphi)$.

Also, the axiom system has the following inference rule to reason about probabilities:

RP. From $\varphi \leftrightarrow \psi$ infer $w(\varphi) = w(\psi)$.

Halpern showed that although this axiom system is sound (i.e. if $\vdash \varphi$ then $\models \varphi$ for every formula φ), there is no sound and complete axiomatization when the domain is not finite. For this reason, we have considered probabilistic languages containing finitely many constants. In this case, the axiom system is sound and complete (i.e. $\vdash \varphi$ iff $\models \varphi$ for every formula φ).

We now give some results from [9] , which are used in the proofs of our theorems. Two formulas φ and ψ are said to be *mutually exclusive* if, from standard first-order reasoning, it follows that $\vdash \neg(\varphi \wedge \psi)$. A set $\varphi_1, \ldots, \varphi_k$ of formulas is mutually exclusive if each pair φ_i, φ_j, for $i \neq j$, is mutually exclusive.

Lemma 1 (1) $\vdash w(true) = 1$.
(2) $\vdash w(false) = 0$.
(3) $\vdash w(\varphi_1 \vee \ldots \vee \varphi_k) = w(\varphi_1) + \cdots + w(\varphi_k)$ *if* $\varphi_1, \ldots, \varphi_k$ *are mutually exclusive.*
(4) *If* $\vdash \varphi$, *then* $\vdash w(\varphi) = 1$.
(5) $\vdash w(\varphi) + w(\neg\varphi) = 1$.

(6) $\vdash w(\varphi \wedge \psi) \leq w(\varphi)$.
(7) $\vdash w(\varphi) \leq w(\varphi \vee \psi)$.
(8) $\vdash w(\varphi) = 1 \rightarrow (w(\varphi \wedge \psi) = w(\psi))$.
(9) $\vdash w(\varphi) = 1 \rightarrow (w(\neg\varphi \wedge \psi) = 0)$.
(10) $\vdash (w(\varphi \leftrightarrow \psi) = 1) \rightarrow (w(\varphi) = w(\psi))$.

3.2 Probabilistic Theories

In this section we show how to formalize probabilistic databases using probabilistic theories. As already said, our work is inspired on Reiter's work on extended relational theories [30] .

Let $\mathcal{L}$ be a two-sorted probabilistic language. A finite set of formulas $\mathcal{T}$ is a *probabilistic theory* iff $\mathcal{T}$ satisfies the following conditions:

(1) For every simple type predicate θ, $\mathcal{T}$ contains exactly one formula of the form

$$(\forall x)(\theta(x) \leftrightarrow x = c^{(1)} \vee \ldots \vee x = c^{(r)}),$$

where $r \geq 0$ and the $c^{(i)}$ are object constants. This formula is called θ's *extension axiom* in $\mathcal{T}$. If $r = 0$, θ's extension axiom is $(\forall x)\neg\theta(x)$.
(2) $\mathcal{T}$ defines a simple type Λ to represent probabilities with the axiom

$$(\forall x^f)(\Lambda(x^f) \leftrightarrow x^f \geq 0 \wedge x^f \leq 1).$$

(3) For every n-ary object predicate P, distinct from equality and simple types, $\mathcal{T}$ contains the formulas:

$$\begin{aligned}
&(\forall \bar{x})(P(\bar{x}) \rightarrow \bar{x} = \bar{c}^{(1)} \vee \ldots \vee \bar{x} = \bar{c}^{(r)} \vee \bar{x} = \bar{d}^{(1)} \vee \ldots \vee \bar{x} = \bar{d}^{(s)}),\\
&(\forall \bar{x})(\bar{x} = \bar{c}^{(1)} \vee \ldots \vee \bar{x} = \bar{c}^{(r)} \rightarrow P(\bar{x})),\\
&(\forall \bar{x})(\forall y^f)(w(P(\bar{x})) = y^f \wedge 0 < y^f < 1 \leftrightarrow (\bar{x} = \bar{d}^{(1)} \wedge y^f = p_1) \vee \ldots\\
&\qquad\qquad \vee (\bar{x} = \bar{d}^{(s)} \wedge y^f = p_s)),
\end{aligned}$$

where $r, s \geq 0$, the $\bar{c}^{(i)}$ and the $\bar{d}^{(i)}$ are distinct tuples of object constants of $\mathcal{L}$, and the p_i are sgf-terms such that $p_i \in]0,1[$. These formulas are called P's *extension axioms* in $\mathcal{T}$. Notice that when P represents a classical relation (i.e. $s = 0$) then P's extension axioms are equivalent to $(\forall \bar{x})(P(\bar{x}) \leftrightarrow \bar{x} = \bar{c}^{(1)} \vee \ldots \vee \bar{x} = \bar{c}^{(r)})$. Finally, if $r + s = 0$, P's extension axiom is $(\forall \bar{x})\neg P(\bar{x})$.
(4) Let $\beta = \{\beta_1, \ldots, \beta_m\}$ be the Herbrand base for the object predicates, i.e., the set of all distinct ground formulas of the form $P(\bar{c})$, where P is an

object predicate and $\bar{c}$ is a tuple of object constants of $\mathcal{L}$. Then, for each subset $\{\beta_{i_1}, \ldots, \beta_{i_k}\} \subseteq \beta$, $k \geq 2$, the theory $\mathcal{T}$ contains the axiom

$$w(\beta_{i_1} \wedge \ldots \wedge \beta_{i_k}) = w(\beta_{i_1}) \times \ldots \times w(\beta_{i_k}).$$

These axioms, called *independence assumption axioms*, are denoted by $IAA_{\mathcal{T}}$. Since $\mathcal{L}$ contains finitely many object constants and predicates, the set β is finite and thus, there are finitely many axioms in $IAA_{\mathcal{T}}$.

(5) $\mathcal{T}$ contains the axiom $(\forall x)(x = x)$, and the axiom $c_i \neq c_j$ for every pair of distinct object constants (c_i, c_j). These axioms are called *unique name axioms* and are denoted by $UNA_{\mathcal{T}}$.

(6) There are no other formulas in $\mathcal{T}$.

Notice that, due to the axioms for simple types, we assume that the domains are finite. As already said, this is needed since no complete axiomatization of first-order probabilistic languages is possible when the domain is infinite.

We next give an example which is used as a running example throughout the paper.

Example 2 *Consider the following database where professor, student, and course are simple types, course_dep is a classical relation, and teaches is a probabilistic relation*

professor
Jean
Paul
Marie

student
Tom
John
Anne

course
Algebra
Calculus
Physics

course_dep

course	*dep*
Algebra	*CS*
Calculus	*CS*
Calculus	*EE*
Physics	*EE*

teaches

professor	*course*	*w*
Jean	*Algebra*	*1.0*
Paul	*Calculus*	*0.7*
Marie	*Calculus*	*0.3*
Marie	*Physics*	*0.9*

Suppose further that the database contains relation takes of Figure 1. The theory $\mathcal{T}$ associated to the database contains, in addition to axioms (1)–(3) for relation takes, the following axioms:

$(\forall x)(professor(x) \leftrightarrow x = Jean \vee x = Paul \vee x = Marie)$,
$(\forall x)(student(x) \leftrightarrow x = Tom \vee x = John \vee x = Anne)$,
$(\forall x)(course(x) \leftrightarrow x = Algebra \vee x = Calculus \vee x = Physics)$,
$(\forall x^f)(\Lambda(x^f) \leftrightarrow x^f \geq 0 \wedge x^f \leq 1)$,
$(\forall x)(\forall y)(course\_dep(x, y) \leftrightarrow$

$$
\begin{array}{l}
\qquad (x = \mathit{Algebra} \wedge y = \mathit{CS}) \vee (x = \mathit{Calculus} \wedge y = \mathit{CS}) \vee \\
\qquad (x = \mathit{Calculus} \wedge y = \mathit{EE}) \vee (x = \mathit{Physics} \wedge y = \mathit{EE})), \\
(\forall x)(\forall y)(\mathit{teaches}(x, y) \rightarrow \\
\qquad (x = \mathit{Jean} \wedge y = \mathit{Algebra}) \vee (x = \mathit{Paul} \wedge y = \mathit{Calculus}) \vee \\
\qquad (x = \mathit{Marie} \wedge y = \mathit{Calculus}) \vee (x = \mathit{Marie} \wedge y = \mathit{Physics})), \\
(\forall x)(\forall y)((x = \mathit{Jean} \wedge y = \mathit{Algebra}) \rightarrow \mathit{teaches}(x, y)), \\
(\forall x)(\forall y)(\forall z^f)(w(\mathit{teaches}(x, y)) = y^f \wedge 0 < y^f < 1 \leftrightarrow \\
\qquad (x = \mathit{Paul} \wedge y = \mathit{Calculus} \wedge z^f = 0.7) \vee \\
\qquad (x = \mathit{Marie} \wedge y = \mathit{Calculus} \wedge z^f = 0.3) \vee \\
\qquad (x = \mathit{Marie} \wedge y = \mathit{Physics} \wedge z^f = 0.9)), \\
w(\mathit{teaches}(\mathit{Jean}, \mathit{Algebra}) \wedge \mathit{teaches}(\mathit{Paul}, \mathit{Calculus})) = \\
\qquad w(\mathit{teaches}(\mathit{Jean}, \mathit{Algebra})) \times w(\mathit{teaches}(\mathit{Paul}, \mathit{Calculus})), \\
\qquad\qquad\qquad \vdots \\
(\forall x)(x = x), \\
\mathit{Jean} \neq \mathit{Paul},\ \mathit{Jean} \neq \mathit{Marie}, \ldots
\end{array}
$$

Theorem 3 *Every probabilistic theory $\mathcal{T}$ is consistent.*

Proof. *We only give the sketch of a proof, which consists in constructing a model of $\mathcal{T}$. Given a probabilistic language $\mathcal{L}$ and a probabilistic theory $\mathcal{T}$, let $P_1, \ldots, P_k$ be the set of object predicates in $\mathcal{L}$ distinct from equality. We associate to every predicate P_i a probabilistic relation of the same name containing the information represented in P_i's extension axioms. To each probabilistic relation P_i we can associate a set of possible worlds $REP(P_i)$. Further, if $\bar{P} = \langle P_1, \ldots, P_l \rangle$, then from $REP(P_i)$ it is easy to construct $REP(\bar{P})$, the set of possible worlds for the predicates of $\bar{P}$ such that each pair $\langle s, p \rangle \in REP(\bar{P})$ denotes a Herbrand interpretation s for the object predicates in $\mathcal{L}$ with its associated probability p. We now prove that $REP(\bar{P})$ defines a model of $\mathcal{T}$. For this, define a structure $M = (\mathcal{D}, S, \pi, \mu)$ as follows. (1) $\mathcal{D}$ is the set of all the object constants of $\mathcal{L}$. (2) The set of states S is such that $s \in S$ iff there is a p such that $\langle s, p \rangle \in REP(\bar{P})$. (3) For every $s \in S$ and for every object constant $a \in \mathcal{L}$, $\pi(s)(a) = a$. (4) For every $s \in S$, $\pi(s)(=)(c, c) = true$ iff $c \in \mathcal{D}$ and false otherwise. (5) For every object predicate P_i and state $s \in S$, $\pi(s)(P_i)(\bar{d}) = true$ iff $P_i(\bar{d}) \in s$ and false otherwise. (6) μ is a discrete probability function on S such that $\mu(s) = p$ iff $\langle s, p \rangle \in REP(\bar{P})$.*

It is simple to verify that M is a model of $\mathcal{T}$. □

Before concluding this section, we recall some notations from [30] . First, the *type-restricted quantifiers* are defined as follows. If τ is a simple type and if φ is a formula, then $(\forall x/\tau)\varphi$ abbreviates $(\forall x)(\tau(x) \rightarrow \varphi)$ and $(\exists x/\tau)\varphi$ abbreviates

$(\exists x)(\tau(x) \wedge \varphi)$. These type-restricted quantifiers restrict the possible x's to just those that belong to domain τ. Also if $\bar{\tau} = \tau_1, \ldots, \tau_n$ is a sequence of simple types and $\bar{c} = (c_1, \ldots, c_n)$ is a tuple of object constants, then $\bar{\tau}(\bar{c})$ denotes the formula $\tau_1(c_1) \wedge \ldots \wedge \tau_n(c_n)$.

3.3 Queries

In a probabilistic language $\mathcal{L}$, queries are expressions of the form

$$Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f) \rangle,$$

where $\bar{x}/\bar{\tau}$ and $\bar{y}^f/\bar{\Lambda}$ denote, respectively, $x_1/\tau_1, \ldots, x_m/\tau_m$ and $y_1^f/\Lambda, \ldots, y_n^f/\Lambda$, the x_i and y_i^f are distinct object and field variables of $\mathcal{L}$, each τ_i is a simple type of $\mathcal{L}$, and $F(\bar{x}, \bar{y}^f)$ is a formula of $\mathcal{L}$ whose free variables are among $\bar{x}$ and $\bar{y}^f$ and whose quantifiers are type-restricted. If $m = n = 0$, queries are of the form $Q = \langle \mid F \rangle$, where F has no free variables and correspond to asking the database if F is true.

Let $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f) \rangle$ be a query and let $\bar{c}$ and $\bar{p}$ be, respectively, tuples of object constants and sgf-terms. Intuitively, $(\bar{c}, \bar{p})$ is an answer to the query Q if $\bar{c}$ and $\bar{p}$ satisfy the simple types $\bar{\tau}$ and $\bar{\Lambda}$, and if $F(\bar{c}, \bar{p})$ is verified in $\mathcal{T}$. In addition, we require $\bar{p}$ to be different from $\bar{0}$ to eliminate unnecessary answers. Formally, $(\bar{c}, \bar{p})$ is an answer to query Q in a probabilistic theory $\mathcal{T}$ if and only if

(1) $\mathcal{T} \vdash \bar{\tau}(\bar{c})$;
(2) $\mathcal{T} \vdash \bar{\Lambda}(\bar{p})$;
(3) $\mathcal{T} \vdash p_i \neq 0$ for at least one $i = 1, \ldots, n$; and
(4) $\mathcal{T} \vdash F(\bar{c}, \bar{p})$.

As usual, the set of answers to a query Q is denoted by $\| Q \|$.

Condition (3) eliminates from $\| Q \|$ those tuples $\bar{c}$ such that $\mathcal{T} \vdash F(\bar{c}, \bar{0})$. For instance, consider the query $Q = \langle x/\tau, y^f/\Lambda \mid w(F(x)) = y^f \rangle$. Since there is always a $p \in [0, 1]$ such that $\mathcal{T} \vdash w(F(c)) = p$, without condition (3) $\| Q \|$ would always contain one answer for each c of the domain τ.

In the special case where the query is of the form $\langle \mid F \rangle$, the null tuple () is the only answer to the query when $\mathcal{T} \vdash F$ and is $\{\}$ otherwise; $\{()\}$ denotes the answer "yes" and $\{\}$ denotes the answer "we don't know". An answer $\{()\}$ to the query $\langle \mid \neg F \rangle$ denotes the answer "no" to the original query $\langle \mid F \rangle$.

For example, consider a probabilistic theory stating that $w(P(a)) = 1$, $w(P(b)) = 0$, and $w(P(c)) = 0.5$. Then, while the answer to $Q_1 = \langle \mid P(a) \rangle$ is

"yes" (since $P(a)$ is true in every possible world), the answer to $Q_2 = \langle \mid P(b) \rangle$ is "we don't know". However, since the answer to $Q'_2 = \langle \mid \neg P(b) \rangle$ is "yes", then the answer to the original query Q_2 is "no". On the contrary, the answer to both $Q_3 = \langle \mid P(c) \rangle$ and $Q'_3 = \langle \mid \neg P(c) \rangle$ is "we don't know".

We give in the next sections a sound and complete algorithm that computes query answers in probabilistic theories. Query evaluation is studied in two stages. First, we study *first-order queries* of the form $Q = \langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f \rangle$, where F is a first-order formula. Then, we study *probabilistic queries* of the form $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f) \rangle$, where F is an arbitrary formula.

For the sake of clarity, we allow the projection, selection, and join operators to use query variables as attributes. For example, consider the queries $Q_1 = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \rangle$ and $Q_2 = \langle \bar{x}/\bar{\tau}, \bar{z}^f/\bar{\Lambda} \mid F_2(\bar{x}, \bar{z}^f) \rangle$, where $\bar{x} = \langle x_1, x_2, x_3 \rangle$, $\bar{y}^f = \langle y_1^f, \ldots, y_4^f \rangle$ and $\bar{z}^f = \langle z_1^f, \ldots, z_5^f \rangle$. Then, the expressions $\pi_{\bar{x}}(\| Q_1 \|)$, $\sigma_{x_2 = c \wedge y_3^f > 0.5}(\| Q_1 \|)$, and $\| Q_1 \| \bowtie_{\bar{x}} \| Q_2 \|$ have their intuitive meaning as follows $\pi_{1,2,3}(\| Q_1 \|)$, $\sigma_{2=c \wedge 6 > 0.5}(\| Q_1 \|)$, and $\| Q_1 \| \bowtie_{1=1 \wedge 2=2 \wedge 3=3} \| Q_2 \|$.

4 First-order queries

As defined above, first-order queries are expressions of the form

$$Q = \langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f \rangle,$$

where F is a first-order formula. The answer to such a query is a set of tuples $(\bar{c}, p)$ such that $\bar{c}$ satisfies the simple types $\bar{\tau}$, $p \in]0, 1]$, and $\mathcal{T} \vdash w(F(\bar{c})) = p$. The answer $\| Q \|$ can be seen as a probabilistic relation.

As shown by the following example, probabilistic relations do not allow to decompose first-order queries in order to obtain the answer to a query from the answer to its subqueries.

Example 4 *Consider the simple type* $\tau = \{a, b, c, d\}$*, the probabilistic relations* P_1*,* P_2*,*

P_1	
(a, b)	0.8
(b, b)	0.7

P_2	
(a, c)	0.7
(a, d)	0.6
(b, c)	0.5

the formula $F = [P_1(x, y) \wedge P_2(y, z)] \vee [P_1(x, y) \wedge P_2(x, z)]$ *and the query* $Q = \langle x/\tau, y/\tau, z/\tau, y^f/\Lambda \mid w(F) = y^f \rangle$.

If $F_1 = P_1(x,y) \wedge P_2(y,z)$ *and* $F_2 = P_1(x,y) \wedge P_2(x,z)$*, the answers to the subqueries* $Q_1 = \langle x/\tau, y/\tau, z/\tau, y^f/\Lambda \mid w(F_1) = y^f \rangle$*, and* $Q_2 = \langle x/\tau, y/\tau, z/\tau, y^f/\Lambda \mid w(F_2) = y^f \rangle$ *are as follows*

$$\| Q_1 \| = \{(a,b,c)/0.8 \times 0.5, (b,b,c)/0.7 \times 0.5\}, \text{ and}$$
$$\| Q_2 \| = \{(a,b,c)/0.8 \times 0.7, (a,b,d)/0.8 \times 0.6, (b,b,c)/0.7 \times 0.5\}.$$

However, in the general case it is not possible to obtain the answer to the original query Q *from the probabilistic relations* $\| Q_1 \|$ *and* $\| Q_2 \|$*. Indeed, since by the axioms of probabilistic logic* $w(F_1 \vee F_2) = w(F_1) + w(F_2) - w(F_1 \wedge F_2)$ *it is necessary to obtain in addition the answer to the query* $Q_3 = \langle x/\tau, y/\tau, z/\tau, y^f/\Lambda \mid w(P_1(x,y) \wedge P_2(y,z) \wedge P_2(x,z)) = y^f \rangle$*.*

After evaluating Q_3*, we obtain the answer to* Q *as follows*

$$\| Q \| = \{(a,b,c)/0.8 \times 0.85, (b,b,c)/0.7 \times 0.5, (a,b,d)/0.8 \times 0.6\}.$$

In order to correctly decompose first-order queries, we define in the next section a particular type of relations called *trace relations*. These relations keep track of the origin of tuples resulting from applying relational operators. Thus, they contain the necessary information to compute the correct probability values from the subqueries of a query. A detailed discussion of these relations is presented in [42] .

4.1 Trace relations

By a *trace relation*, briefly a *t-relation*, we mean a classical relation extended with one additional special column, called *trace*, containing for every tuple a formula that traces the information of how the tuple has been obtained.

Definition 5 *Given a probabilistic theory* $\mathcal{T}$*, the set of formulas* $\mathcal{F}_\mathcal{T}$ *is formed by starting with true, false, and* $P(\bar{c})$ *where* P *is an object predicate and* $\bar{c}$ *is a tuple of object constants, and closing off under conjunction, disjunction, and negation* [2] *. If* F_1 *and* F_2 *are formulas from* $\mathcal{F}_\mathcal{T}$*, then* F_1F_2 *denotes the conjunction of both formulas and* $\overline{F}_1$ *denotes* $\neg F_1$*.*

Definition 6 *Let* $\mathcal{R}(A_1, \ldots, A_n)$ *be a relation scheme, where* $\mathrm{dom}(A_i)$ *is the domain of* A_i*, for* $i = 1, \ldots, n$*. Then, a t-relation* R *on* $\mathcal{R}$ *is defined as follows:*

$$R \subset \{\bar{c}/\varphi \mid \bar{c} = (c_1, \ldots, c_n) \in \mathrm{dom}(A_1) \times \ldots \times \mathrm{dom}(A_n) \wedge \varphi \in \mathcal{F}_\mathcal{T}\}.$$

[2] Formulas in $\mathcal{F}_\mathcal{T}$ should be considered as propositional formulas. In fact, it is equivalent to construct $\mathcal{F}_\mathcal{T}$ by assigning a unique propositional constant p_i to every atom of the Herbrand base β for the object predicates.

For a tuple $\bar{c}/\varphi$, we say that $\bar{c}$ is the pure tuple and φ is the trace attribute.

A t-relation R is represented either in set notation as $R = \{\bar{c}_1/\varphi_1, \ldots, \bar{c}_m/\varphi_m\}$ or in a tabular form where the trace attribute is represented in an additional column. An example of t-relation is given below.

abc	$P(ab) \wedge [P(ac) \vee Q(ac)]$
abb	$P(ab)$
acd	$[P(ac) \vee Q(ac)] \wedge [P(ad) \vee Q(cd)]$

T-relations have some similarities with Assumption-Based Truth Maintenance Systems (e.g., [31]). In fact, a tuple $\bar{c}/\varphi$ in a t-relation R represents the assertion "$R(\bar{c})$ *is true in all the possible worlds in which* φ *is true*". Thus, φ is the disjunction of all the justifications of $R(\bar{c})$. As it follows from the definition of relational operators (given later in this section), t-relations allow to compute, in an algebraic way, the set of justifications for every first-order formula F and tuple $\bar{c}$. Notice that the concept of t-relations have also been studied in [34,33,16,15] . T-relations have also some similarities with the C-tables of [11] .

T-relations can contain two types of redundancies. First, a tuple $\bar{c}/\varphi$ can be such that φ is equivalent to *false*; in this case the tuple can be eliminated. Second, a t-relation can contain a set of tuples $\{\bar{c}/\varphi_1, \ldots, \bar{c}/\varphi_n\}$; this redundancy is eliminated by replacing the set of tuples with $\bar{c}/\varphi$ where $\varphi = \varphi_1 \vee \ldots \vee \varphi_n$.

We now define an operator, called *REDUCE*, that takes as argument a t-relation R and gives as result a t-relation R^0 obtained by removing every redundancy from R.

Definition 7 *Let R be a t-relation. We define $REDUCE(R) = R^0$ where*

$$R^0 = \{\bar{c}/\varphi \mid \textit{for } n \geq 1, \{\bar{c}/\varphi_1, \ldots, \bar{c}/\varphi_n\} \subseteq R \textit{ are all the tuples having } \bar{c} \textit{ as pure tuple and } \varphi = \varphi_1 \vee \ldots \vee \varphi_n \wedge \neg(\varphi \leftrightarrow \textit{false})\}.$$

Relational operators over t-relations are similar to classical relational operators. Redundancies are avoided with the *REDUCE* operator.

Definition 8 *(Projection) If R_1 is a t-relation of scheme $\mathcal{R}_1(\bar{A}, \bar{B})$, then $\pi_{\bar{A}}(R_1) = REDUCE(R)$ where*

$$R = \{\bar{a}/\varphi \mid (\exists \bar{b})((\bar{a}, \bar{b}/\varphi \in R_1)\}.$$

Definition 9 *(Selection) If R_1 is a t-relation and H a selection formula, then $\sigma_H(R_1) = REDUCE(R)$ where*

$$R = \{\bar{c}/\varphi \mid \bar{c}/\varphi \in R_1 \wedge H(\bar{c})\},$$

and $H(\bar{c})$ is the formula H in which the number of attribute i is replaced by c_i.

Definition 10 *(Union) If R_1, R_2 are two domain-compatible t-relations, then $R_1 \cup R_2 = REDUCE(R)$ where*

$$R = \{\bar{c}/\varphi \mid \bar{c}/\varphi \in R_1 \vee \bar{c}/\varphi \in R_2\}.$$

Definition 11 *(Difference) If R_1 and R_2 are two domain-compatible t-relations, then $R_1 - R_2 = REDUCE(R)$ where*

$$R = \{\bar{c}/\varphi \,|\, \bar{c}/\varphi_1 \in R_1 \wedge ([\bar{c}/\varphi_2 \in R_2 \wedge \varphi = \varphi_1 \wedge \neg\varphi_2] \vee [\bar{c}/\phi \notin R_2 \text{ for any } \phi \wedge \varphi = \varphi_1])\}.$$

Definition 12 *(Intersection) If R_1 and R_2 are two domain-compatible t-relations, then $R_1 \cap R_2 = REDUCE(R)$ where*

$$R = \{\bar{c}/\varphi \mid \bar{c}/\varphi_1 \in R_1 \wedge \bar{c}/\varphi_2 \in R_2 \wedge \varphi = \varphi_1 \wedge \varphi_2\}.$$

Definition 13 *(Cartesian product) If R_1 and R_2 are two t-relations, then $R_1 \times R_2 = REDUCE(R)$ where*

$$R = \{\bar{a}\bar{b}/\varphi \mid \bar{a}/\varphi_1 \in R_1 \wedge \bar{b}/\varphi_2 \in R_2 \wedge \varphi = \varphi_1 \wedge \varphi_2\}.$$

Definition 14 *(Division) Let R_1 and R_2 be t-relations of scheme $\mathcal{R}_1(\bar{A}, \bar{B})$ and $\mathcal{R}_2(\bar{B})$ respectively, where $R_2 = \{\bar{b}_1/\psi_1, \ldots, \bar{b}_n/\psi_n\}$. Then $R_1 \div R_2 = REDUCE(R)$ where*

$$R = \{\bar{a}/\varphi \,|\, (\forall i)(1 \leq i \leq n \rightarrow [\bar{a}\bar{b}_i/\varphi_i \in R_1] \vee [\varphi_i = false]) \wedge \varphi = \bigwedge_{i=1}^{n} \psi_i \rightarrow \varphi_i)\}.$$

Notice that it is required that either (1) $\bar{a}\bar{b}_i/\varphi_i$ belongs to R_1 or (2) the pure tuple $\bar{a}\bar{b}_i$ does not appear in R_1, which implicitly means that $\bar{a}\bar{b}_i/$ false belongs to R_1.

The intuition for the term $\psi_i \rightarrow \varphi_i$ is as follows. Since $\bar{b}_i \in R_2$ when ψ_i is true, then $\bar{a} \in R_1 \div R_2$ provided that when ψ_i is true, φ_i is also true.

Notice that if R_2 corresponds to a classical relation, i.e. $R_2 = \{\bar{b}_1/\, true, \ldots, \bar{b}_n/\, true\}$, then in the above definition φ is given by $\varphi = \bigwedge_{i=1}^{n} \varphi_i$.

The trace relational algebra defined above is similar to the "information source tracking" proposed in [33] except for division which is not defined there. We studied in [42] the semantical correctness of these algebraic operators and

proved that all operators but join and Cartesian product satisfy a strong correctness criteria, whereas these two operators satisfy a weak correctness criteria.

Example 15 *Given the following t-relation* R^t

R^t

abd	$R(abd)$
abe	$R(abe)$
abf	$R(abf)$
ace	$R(ace)$
bce	$R(bce)$
ccf	$R(ccf)$

let us evaluate the expression $f(R^t) = \sigma_{A=a \vee A=b}(\pi_{AC}(\pi_{AB}(R^t) \bowtie \pi_{BC}(R^t)))$.

The t-relations $S = \pi_{AB}(R^t)$ *and* $T = \pi_{BC}(R^t)$ *are as follows.*

S

ab	$R(abd) \vee R(abe) \vee R(abf)$
ac	$R(ace)$
bc	$R(bce)$
cc	$R(ccf)$

T

bd	$R(abd)$
be	$R(abe)$
bf	$R(abf)$
ce	$R(ace) \vee R(bce)$
cf	$R(ccf)$

Then $U = S \bowtie T$ *and* $\pi_{AC}(U)$ *are given below.*

U

abd	$R(abd)$
abe	$R(abe)$
abf	$R(abf)$
ace	$R(ace)$
acf	$R(ace) \wedge R(ccf)$
bce	$R(bce)$
bcf	$R(bce) \wedge R(ccf)$
cce	$R(ccf) \wedge [R(ace) \vee R(bce)]$
ccf	$R(ccf)$

$\pi_{AC}(U)$

ad	$R(abd)$
ae	$R(abe) \vee R(ace)$
af	$[R(abf) \vee R(ace)] \wedge$
	$[R(abf) \vee R(ccf)]$
be	$R(bce)$
ce	$R(ccf) \wedge [R(ace) \vee R(bce)]$
cf	$R(ccf)$

Finally, $f(R^T)$ *is given below.*

$f(R^t)$

ad	$R(abd)$
ae	$R(abe) \vee R(ace)$
af	$[R(abf) \vee R(ace)] \wedge [R(abf) \vee R(ccf)]$
be	$R(bce)$

Consider now a first-order query of the form $Q = \langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f \rangle$. To evaluate it we associate to each object predicate a t-relation, and we associate to Q a t-query Q^t (defined below). The answer to Q^t is obtained by applying (extended) algebraic operators to the t-relations. The answer to the original query Q is thus obtained by transforming the t-relation $\| Q^t \|$ into a probabilistic relation $\| Q \|$, that is, by replacing a tuple $\bar{c}/\varphi$ in $\| Q^t \|$ by a tuple $(\bar{c}, p)$ in $\| Q \|$ where $\mathcal{T} \vdash w(F(\bar{c})) = w(\varphi) = p$. All this is formalized in the following sections.

4.2 T-queries

In a probabilistic language $\mathcal{L}$, *t-queries* are expressions of the form $Q^t = \langle \bar{x}/\bar{\tau} \mid F(\bar{x}) \rangle$, where $F(\bar{x})$ is a first-order formula of $\mathcal{L}$ whose free variables are among $\bar{x}$ and whose quantifiers are type-restricted. If F has no free variables, the query is of the form $Q^t = \langle \mid F \rangle$.

Let $\bar{c}$ be a tuple of object constants and φ be a formula from $\mathcal{F}_\mathcal{T}$. Then, $\bar{c}/\varphi$ is an answer to the t-query $Q^t = \langle \bar{x}/\bar{\tau} \mid F(\bar{x}) \rangle$ in a probabilistic theory $\mathcal{T}$ if and only if

(1) $\mathcal{T} \vdash \bar{\tau}(\bar{c})$;
(2) $\mathcal{T} \vdash w(F(\bar{c})) > 0$;
(3) $\mathcal{T} \vdash \varphi \leftrightarrow F(\bar{c})$; and
(4) No atom P in φ is such that $\mathcal{T} \vdash P$ or $\mathcal{T} \vdash \neg P$.

Since there are many formulas φ' in $\mathcal{F}_\mathcal{T}$ satisfying condition (3), condition (4) selects the most general of them, i.e. the formula φ containing the least number of literals. Thus, given a t-query Q^t and a tuple $\bar{c}$ of constants, there is only one formula satisfying $\bar{c}/\varphi \in \| Q^t \|$. This is shown in the following example.

Example 16 *Consider a probabilistic theory $\mathcal{T}$ and a t-query*

$$Q^t = \langle x/\tau \mid (\exists y/\tau) P(x, y) \rangle.$$

Suppose that $\mathcal{T}$ defines the simple type $\tau = \{a, b, c\}$ and the probabilistic relation $P = \{(a, a)/0.5, (a, b)/0.6, (b, b)/1.0\}$ with the following extension axioms:

$$\begin{aligned}
&(\forall x)(\tau(x) \leftrightarrow x = a \vee x = b \vee x = c),\\
&(\forall x)(\forall y)(P(x, y) \rightarrow (x = a \wedge y = a) \vee\\
&\qquad (x = a \wedge y = b) \vee (x = b \wedge y = b)),\\
&(\forall x)(\forall y)(x = b \wedge y = b \rightarrow P(x, y)),
\end{aligned}$$

$$(\forall x)(\forall y)(\forall z^f)(w(P(x,y)) = z^f \wedge 0 < z^f < 1 \leftrightarrow$$
$$(x = a \wedge y = a \wedge z^f = 0.5) \vee (x = a \wedge y = b \wedge z^f = 0.6)).$$

By τ's extension axiom, we have $\mathcal{T} \vdash (\exists y/\tau)P(x,y) \leftrightarrow P(x,a) \vee P(x,b) \vee P(x,c)$.

Consider now constant a. By P's first extension axiom, $\mathcal{T} \vdash \neg P(a,c)$ *and then* $\mathcal{T} \vdash P(a,a) \vee P(a,b) \vee P(a,c) \leftrightarrow P(a,a) \vee P(a,b)$. *Conditions (1)–(2) are verified since* $\mathcal{T} \vdash \tau(a)$ *and* $\mathcal{T} \vdash w((\exists y/\tau)P(x,y)) > 0$. *Also, since* $P(a,a) \vee P(a,b)$ *satisfies conditions (3)–(4), then* $a/P(a,a) \vee P(a,b) \in \| Q^t \|$.

Consider now constant b. Since $\mathcal{T} \vdash P(b,b)$, *then* $\mathcal{T} \vdash (\exists y/\tau)P(b,y) \leftrightarrow true$. *Since* $\mathcal{T} \vdash \tau(b)$ *and* $\mathcal{T} \vdash w((\exists y/\tau)P(b,y)) > 0$, *then b/ true* $\in \| Q^t \|$.

In a probabilistic theory $\mathcal{T}$, we associate to every object predicate P a t-relation $|\, P \,|^t$ as follows. If P is an object predicate whose extension axiom is $(\forall \bar{x})\neg P(\bar{x})$, then $|\, P \,|^t = \{\}$. If θ is a simple type whose extension axiom is

$$(\forall x)(\theta(x) \leftrightarrow x = c^{(1)} \vee \ldots \vee x = c^{(r)}),$$

then $|\, \theta \,|^t = \{c^{(1)}/\, true, \ldots, c^{(r)}/\, true\}$. If P is an object predicate, different from equality and from simple types, whose first two extension axioms in $\mathcal{T}$ are

$$(\forall \bar{x})(P(\bar{x}) \rightarrow \bar{x} = \bar{c}^{(1)} \vee \ldots \vee \bar{x} = \bar{c}^{(r)} \vee \bar{x} = \bar{d}^{(1)} \vee \ldots \vee \bar{x} = \bar{d}^{(s)}),$$
$$(\forall \bar{x})(\bar{x} = \bar{c}^{(1)} \vee \ldots \vee \bar{x} = \bar{c}^{(r)} \rightarrow P(\bar{x})),$$

then $|\, P \,|^t = \{\bar{c}^{(1)}/\, true, \ldots, \bar{c}^{(r)}/\, true, \bar{d}^{(1)}/P(\bar{d}^{(1)}), \ldots, \bar{d}^{(s)}/P(\bar{d}^{(s)})\}$. Also, for the object equality we define

$$|=|^t \stackrel{\text{def}}{=} \{(c,c)/\, true \mid c \text{ is an object constant of } \mathcal{L}\}.$$

Now, let $\bar{\tau} = \langle \tau_1, \ldots, \tau_n \rangle$ be a sequence of simple types. If $n = 0$, then $|\, \bar{\tau} \,|^t$ denotes $\{()\}$, and if $n > 0$, then

$$|\, \bar{\tau} \,|^t = \{\bar{c}/\, true \mid \bar{c} = (c_1, \ldots, c_n) \wedge (\forall i)(1 \leq i \leq n \rightarrow c_i/\, true \in |\, \tau_i \,|^t)\}.$$

Before studying the evaluation of t-queries, we give some preliminary lemmas. The easy proofs are omitted.

Lemma 17 *Let $\mathcal{T}$ be a probabilistic theory, let $\bar{\tau}$ be a sequence of simple types, and let $\bar{c}$ be a tuple of object constants. Then* $\mathcal{T} \vdash \bar{\tau}(\bar{c})$ *iff* $\bar{c}/\, true \in |\, \bar{\tau} \,|^t$. $\square$

The next lemma relates the probability of atomic formulas of the form $P(\bar{c})$ in a probabilistic theory $\mathcal{T}$ with the t-relation $|P|^t$.

Lemma 18 *Let $\mathcal{T}$ be a probabilistic theory, let P be an object predicate, possibly a simple type or equality, and let $\bar{c}$ be a tuple of object constants. Then*

(1) $\mathcal{T} \vdash w(P(\bar{c})) > 0$ *iff either* $\bar{c}/$ *true or* $\bar{c}/P(\bar{c})$ *belongs to* $|P|^t$.
(2) $\mathcal{T} \vdash w(P(\bar{c})) < 1$ *iff* $\bar{c}/$ *true* $\notin |P|^t$. $\square$

Given the independence axioms in probabilistic theories, the next lemma allows to recursively decompose a complex query into simpler subqueries. This lemma is used with queries containing conjunctions and universal quantifiers.

Lemma 19 *Let $\mathcal{T}$ be a probabilistic theory and let F_1 and F_2 be ground first-order formulas without quantifiers. Then $\mathcal{T} \vdash w(F_1) > 0$ and $\mathcal{T} \vdash w(F_2) > 0$ iff $\mathcal{T} \vdash w(F_1 \wedge F_2) > 0$.* $\square$

The next lemma, combined with the axioms in probabilistic languages, tells us that a first-order formula F has probability 0 in a probabilistic theory $\mathcal{T}$ iff $\mathcal{T} \vdash \neg F$.

Lemma 20 *Let $\mathcal{T}$ be a probabilistic theory, let $F(\bar{x})$ be a first-order formula with type-restricted quantifiers and let $\bar{c}$ be a tuple of object constants. Then $\mathcal{T} \vdash w(F(\bar{c})) = 0$ iff $\mathcal{T} \vdash \neg F(\bar{c})$.* $\square$

Finally, the next lemma states that if a pure tuple $\bar{c}$ does not appear in the answer of a t-query Q^t, then it has probability 0 to satisfy the associated query Q.

Lemma 21 *Let $\mathcal{T}$ be a probabilistic theory, let $Q^t = \langle \bar{x}/\bar{\tau} \mid F(\bar{x}) \rangle$ be a t-query and let $\bar{c}$ be a tuple of object constants such that $\mathcal{T} \vdash \bar{\tau}(\bar{c})$. Then there $\bar{c}/\varphi \notin \| Q^t \|$ for no formula φ iff $\mathcal{T} \vdash w(F(\bar{c})) = 0$.* $\square$

4.3 Primitive t-queries

This section shows how to evaluate primitive queries of the form $\langle \bar{x}/\bar{\tau} \mid P(\bar{r}) \rangle$ or of the form $\langle \bar{x}/\bar{\tau} \mid \neg P(\bar{r}) \rangle$ where P is an object predicate or the equality. But prior to that, we give preliminary definitions.

Definition 22 *[30] Let $m \geq n$, let $\bar{r}$ be a m-tuple of variables and/or constants, let $\bar{x} = x_1, \ldots, x_n$ be a sequence of distinct variables where each x_i is a variable that appears in $\bar{r}$ and let $\bar{c} = (c_1, \ldots, c_n)$ be a tuple of constants. We define $\bar{r}_{\bar{c}|\bar{x}}$ as the m-tuple obtained replacing in $\bar{r}$ each occurrence of x_i by c_i, for $i = 1, \ldots, n$.*

For example, $(x, y, a, x, z, y)_{(b,c,d)|(y,x,z)} = (c, b, a, c, d, b)$.

Definition 23 *[38] Let $\bar{r} = (r_1, \ldots, r_m)$ be an m-tuple of variables and/or constants and let $\bar{x} = (x_1, \ldots, x_n)$ be a sequence of distinct variables where each x_i appears in $\bar{r}$. We define $F(\bar{r}, \bar{x})$ as the conjunction of formulas of the form: (1) $i = r_i$ if r_i is a constant and (2) $i = j$ if r_i is a variable, for example x_k, and if r_j is an occurrence of x_k where $1 \leq j \leq m$.*

For example, if $\bar{r} = x, y, a, x, z, y$ and $\bar{x} = x, y, z$ then $F(\bar{r}, \bar{x})$ is $1 = 4 \wedge 2 = 6 \wedge 3 = a$.

The following theorem shows how to obtain the answer to primitive t-queries of the form $\langle \bar{x}/\bar{\tau} \mid P(\bar{r}) \rangle$ where P is an object predicate.

Theorem 24 *Let $\mathcal{T}$ be a probabilistic theory and let $\langle \bar{x}/\bar{\tau} \mid P(\bar{r}) \rangle$ be a primitive t-query where P is an object predicate, let $\bar{x} = (x_1, \ldots, x_n)$, and let $\bar{r} = (r_1, \ldots, r_m)$ is a m-tuple of object constants and/or variables from $x_1, \ldots, x_n$. Suppose further for $j = 1, \ldots, n$, that r_{i_j} is the first occurrence of x_j in r. Then*

$$\{\bar{x}/\bar{\tau} \mid P(\bar{r})\}^t \;=\; |\,\bar{\tau}\,|^t \;\cap\; \pi_{i_1 \ldots i_n} \sigma_{F(\bar{r},\bar{x})}(|\,P\,|^t). \tag{4}$$

Proof. *Let $Q^t = \langle \bar{x}/\bar{\tau} \mid P(\bar{r}) \rangle$. Then $\bar{c}/\varphi$ belongs to the left-hand side of (4) iff*

(1) $\mathcal{T} \vdash \bar{\tau}(\bar{c})$;
(2) $\mathcal{T} \vdash w(P(\bar{r}_{\bar{c}|\bar{x}})) > 0$;
(3) $\mathcal{T} \vdash \varphi \leftrightarrow P(\bar{r}_{\bar{c}|\bar{x}})$; *and*
(4) *There is no atom P in φ such that $\mathcal{T} \vdash P$ or $\mathcal{T} \vdash \neg P$.*

By Lemma 17, $\mathcal{T} \vdash \bar{\tau}(\bar{c})$ iff $\bar{c}/$ true $\in |\,\bar{\tau}\,|^t$. By Lemma 18, (2) is verified iff either $\bar{r}_{\bar{c}|\bar{x}}/$ true or $\bar{r}_{\bar{c}|\bar{x}}/P(\bar{r}_{\bar{c}|\bar{x}})$ belongs to $|\,P\,|^t$. Also, by Lemma 20, (2) is verified iff $\mathcal{T} \nvdash \neg P(\bar{r}_{\bar{c}|\bar{x}})$.

Notice that $\bar{r}_{\bar{c}|\bar{x}}/\varphi \in |\,P\,|^t$ iff $\bar{r}_{\bar{c}|\bar{x}}/\varphi \in \sigma_{F(\bar{r},\bar{x})}(|\,P\,|^t)$ iff $\bar{c}/\varphi \in \pi_{i_1 \ldots i_n} \sigma_{F(\bar{r},\bar{x})}(|\,P\,|^t)$. Also, notice that φ cannot be equal to false since in that case, by (3), it follows that $\mathcal{T} \vdash \neg P(\bar{r}_{\bar{c}|\bar{x}})$ and $\mathcal{T} \vdash w(P(\bar{r}_{\bar{c}|\bar{x}})) = 0$, contradicting (2). Hence, (3) and (4) are verified iff either $\varphi =$ true or $\varphi = P(\bar{r}_{\bar{c}|\bar{x}})$.

If $\varphi =$ true, then $\mathcal{T} \vdash P(\bar{r}_{\bar{c}|\bar{x}})$ and $\mathcal{T} \vdash w(P(\bar{r}_{\bar{c}|\bar{x}})) = 1$. Notice that $\bar{r}_{\bar{c}|\bar{x}}/P(\bar{r}_{\bar{c}|\bar{x}}) \notin |\,P\,|^t$, since in that case, P is an object predicate whose third extension axiom is

$$(\forall \bar{x})(\forall y^f)(w(P(\bar{x})) = y^f \wedge 0 < y^f < 1 \leftrightarrow (\bar{x} = \bar{d}^{(1)} \wedge y^f = p_1) \vee \ldots$$

$$\vee\ (\bar{x} = \bar{d}^{(s)} \wedge y^f = p_s)),$$

contradiction. Therefore, $\bar{r}_{\bar{c}|\bar{x}}$*/ true* $\in |\,P\,|^t$*,* $\bar{c}$*/ true* $\in \pi_{i_1 \ldots i_n}\sigma_{F(\bar{r},\bar{x})}(|\,P\,|^t)$*, and then* $\bar{c}$*/ true belongs to the right-hand side of (4).*

If $\varphi = P(\bar{r}_{\bar{c}|\bar{x}})$*, then* $\mathcal{T} \nvdash P(\bar{r}_{\bar{c}|\bar{x}})$ *and by (2),* $\mathcal{T} \nvdash \neg P(\bar{r}_{\bar{c}|\bar{x}})$*. Thus, P cannot be the equality or a simple type because in those cases, either* $\mathcal{T} \vdash P(\bar{r}_{\bar{c}|\bar{x}})$ *or* $\mathcal{T} \vdash \neg P(\bar{r}_{\bar{c}|\bar{x}})$*. Thus, P is an object predicate whose extension axioms are*

$$\begin{aligned}
&(\forall \bar{x})(P(\bar{x}) \rightarrow \bar{x} = \bar{c}^{(1)} \vee \ldots \vee \bar{x} = \bar{c}^{(r)} \vee \bar{x} = \bar{d}^{(1)} \vee \ldots \vee \bar{x} = \bar{d}^{(s)}),\\
&(\forall \bar{x})(\bar{x} = \bar{c}^{(1)} \vee \ldots \vee \bar{x} = \bar{c}^{(r)} \rightarrow P(\bar{x})),\\
&(\forall \bar{x})(\forall y^f)(w(P(\bar{x})) = y^f \wedge 0 < y^f < 1 \leftrightarrow (\bar{x} = \bar{d}^{(1)} \wedge y^f = p_1) \vee \ldots\\
&\qquad\qquad \vee\ (\bar{x} = \bar{d}^{(s)} \wedge y^f = p_s)),
\end{aligned}$$

By standard equality reasoning, since $\mathcal{T} \nvdash P(\bar{r}_{\bar{c}|\bar{x}})$ *and* $\mathcal{T} \nvdash \neg P(\bar{r}_{\bar{c}|\bar{x}})$*, then* $\mathcal{T} \vdash \bar{r}_{\bar{c}|\bar{x}} = \bar{d}^{(1)} \vee \ldots \vee \bar{r}_{\bar{c}|\bar{x}} = \bar{d}^{(s)}$*. Hence,* $\bar{r}_{\bar{c}|\bar{x}}/P(\bar{r}_{\bar{c}|\bar{x}}) \in |\,P\,|^t$ *and* $\bar{c}/P(\bar{r}_{\bar{c}|\bar{x}})$ *belongs to the right-hand side of (4).* □

For example, the above theorem states that the answer to $Q^t = \langle x/\tau, y/\theta \mid P(a, x, y, x)\rangle^t$ is given by $\|\,Q^t\,\| = (\,|\,\tau\,|^t \times |\,\theta\,|^t\,) \cap \pi_{2,3}\sigma_{1=a\wedge 2=4}(|\,P\,|^t)$.

The following theorem shows how to obtain the answer to primitive t-queries of the form $\langle \bar{x}/\bar{\tau} \mid \neg P(\bar{r})\rangle$ where P is an object predicate.

Theorem 25 *Let* $\mathcal{T}$ *be a probabilistic theory and let* $\langle \bar{x}/\bar{\tau} \mid \neg P(\bar{r})\rangle$ *be a primitive t-query where P is an object predicate,* $\bar{x} = (x_1, \ldots, x_n)$*, and* $\bar{r} = (r_1, \ldots, r_m)$ *is a m-tuple of object constants and/or variables from* $x_1, \ldots, x_n$*. Suppose further for* $j = 1, \ldots, n$*, that* r_{i_j} *is the first occurrence of* x_j *in r. Then*

$$\{\bar{x}/\bar{\tau} \mid \neg P(\bar{r})\}^t \;=\; |\,\bar{\tau}\,|^t \;-\; \pi_{i_1 \ldots i_n}\sigma_{F(\bar{r},\bar{x})}(|\,P\,|^t). \tag{5}$$

Proof. *Similar to the proof of Theorem 24.* □

For example, the above theorem is used for answering the query

$$\langle s/\mathrm{stud}, p/\Lambda \mid w(\neg \mathrm{takes}(s, \mathrm{Algebra})) = p\rangle$$

which asks for the tuples $\langle s, p\rangle$ such that p is the probability that student t does not take the course of Algebra.

For primitive t-queries involving the object equality, we have similar results as in [30] .

Theorem 26 *Let $\mathcal{T}$ be a probabilistic theory and let a and b be two constants. Then*

- $\{ \mid a = b\}^t = \{()\}$ *if a and b are identical constants,*
  $= \{\}$ *otherwise.*
- $\{x/\tau \mid x = x\}^t = |\tau|^t$.
- $\{x/\tau \mid x = a\}^t = \{a/\ true\}$ *if $a/\ true \in |\tau|^t$,*
  $= \{\}$ *otherwise.*
- $\{x/\tau, y/\theta \mid x = y\}^t = \{(c,c)/\ true \mid\ c/\ true \in |\tau|^t \wedge c/\ true \in |\theta|^t\}$.
- $\{ \mid a \neq b\} = \{()\}$ *if a and b are distinct constants,*
  $= \{\}$ *otherwise.*
- $\{x/\tau \mid x \neq x\}^t = \{\}$.
- $\{x/\tau \mid x \neq a\}^t = |\tau|^t - \{a/\ true\}$.
- $\{x/\tau, y/\theta \mid x \neq y\}^t = \{(a,b)/\ true \mid\ a/\ true \in |\tau|^t \wedge\ b/\ true \in |\theta|^t$
  and a and b are distinct constants$\}$. □

4.4 Compound t-queries

The next two theorems allow to recursively decompose t-queries containing conjunctions and disjunctions.

Theorem 27 *If $\mathcal{T}$ is a probabilistic theory and if F_1, F_2 are first-order formulas with type-restricted quantifiers, then*

$$\{\bar{x}/\bar{\tau} \mid F_1(\bar{x}) \wedge F_2(\bar{x})\}^t = \{\bar{x}/\bar{\tau} \mid F_1(\bar{x})\}^t \cap \{\bar{x}/\bar{\tau} \mid F_2(\bar{x})\}^t. \tag{6}$$

Proof. *Consider $Q^t = \langle \bar{x}/\bar{\tau} \mid F_1(\bar{x}) \wedge F_2(\bar{x})\rangle$ and its subqueries $Q_1^t = \langle \bar{x}/\bar{\tau} \mid F_1(\bar{x})\rangle$, and $Q_2^t = \langle \bar{x}/\bar{\tau}, \mid F_2(\bar{x})\rangle$. By definition of intersection in t-relations, $\bar{c}/\varphi$ belongs to the right-hand side of (6) iff $\bar{c}/\varphi_1 \in \| Q_1^t \|$, $\bar{c}/\varphi_2 \in \| Q_2^t \|$ and $\varphi = \varphi_1 \wedge \varphi_2$. Thus we have $\mathcal{T} \vdash \bar{\tau}(\bar{c})$, we have*

(1a) $\mathcal{T} \vdash w(F_1(\bar{c})) > 0$, *(1b)* $\mathcal{T} \vdash w(F_2(\bar{c})) > 0$,
(2a) $\mathcal{T} \vdash \varphi_1 \leftrightarrow F_1(\bar{c})$, *(2b)* $\mathcal{T} \vdash \varphi_2 \leftrightarrow F_2(\bar{c})$,

and there is no atom P in φ_1 or in φ_2 such that $\mathcal{T} \vdash P$ or $\mathcal{T} \vdash \neg P$. The last condition is obviously verified for the formula $\varphi_1 \wedge \varphi_2$. Furthermore, by Lemma 19, (1a) and (1b) are verified iff $\mathcal{T} \vdash w(F_1(\bar{c}) \wedge F_2(\bar{c})) > 0$. Finally, by standard first-order reasoning, (2a) and (2b) are verified iff $\mathcal{T} \vdash \varphi_1 \wedge \varphi_2 \leftrightarrow F_1(\bar{c}) \wedge F_2(\bar{c})$ and we arrive at the result. □

For example, the above theorem is used for answering the query

$$\langle s/\text{stud}, p/\Lambda \mid w(\text{takes}(s, \text{Algebra}) \wedge (\exists c/\text{course})(\text{teaches}(\text{Marie}, c) \wedge \text{takes}(s, c))) = p\rangle$$

which asks for the tuples $\langle s, p\rangle$ such that p is the probability that student s takes the course of Algebra and at least one course teached by Marie.

Theorem 28 *If $\mathcal{T}$ is a probabilistic theory and if F_1, F_2 are first-order formulas with type-restricted quantifiers, then*

$$\{\bar{x}/\bar{\tau} \mid F_1(\bar{x}) \vee F_2(\bar{x})\}^t = \{\bar{x}/\bar{\tau} \mid F_1(\bar{x})\}^t \cup \{\bar{x}/\bar{\tau} \mid F_2(\bar{x})\}^t. \tag{7}$$

Proof. *Consider $Q^t = \langle \bar{x}/\bar{\tau} \mid F_1(\bar{x}) \vee F_2(\bar{x})\rangle$ and its subqueries $Q_1^t = \langle \bar{x}/\bar{\tau} \mid F_1(\bar{x})\rangle$, and $Q_2^t = \langle \bar{x}/\bar{\tau} \mid F_2(\bar{x})\rangle$. By definition of union in t-relations, $\bar{c}/\varphi$ belongs to the right-hand side of (7) iff one of the following cases is verified:*

(1) *there is a formula φ_1 such that $\bar{c}/\varphi_1 \in \|Q_1\|^t$ but there is no $\bar{c}/\varphi_2 \in \|Q_2\|^t$ and $\varphi = \varphi_1$;*
(2) *there is a formula φ_2 such that $\bar{c}/\varphi_2 \in \|Q_2\|^t$ but there is no $\bar{c}/\varphi_1 \in \|Q_1\|^t$ and $\varphi = \varphi_2$; or*
(3) *there are formulas φ_1, φ_2 such that $\bar{c}/\varphi_1 \in \|Q_1\|^t$, $\bar{c}/\varphi_2 \in \|Q_2\|^t$ and $\varphi = \varphi_1 \vee \varphi_2$.*

In all the cases we have $\mathcal{T} \vdash \bar{\tau}(\bar{c})$. Let us analyze (1). We have $\mathcal{T} \vdash \varphi_1 \leftrightarrow F_1(\bar{c})$, $\mathcal{T} \vdash w(F_1(\bar{c})) > 0$ and, by Lemma 1, $\mathcal{T} \vdash w(F_1(\bar{c}) \vee F_2(\bar{c})) > 0$. Since by Lemma 21, $\mathcal{T} \vdash w(F_2(\bar{c})) = 0$, by Lemma 20, $\mathcal{T} \vdash \neg F_2(\bar{c})$, i.e., $\mathcal{T} \vdash \mathit{false} \leftrightarrow F_2(\bar{c})$. Thus, $\mathcal{T} \vdash \varphi \leftrightarrow F_1(\bar{c}) \vee F_2(\bar{c})$ and the result follows. The proof for (2) is similar.

Let us analyze case (3). Since $\bar{c}/\varphi_1 \in \|Q_1^t\|$, $\bar{c}/\varphi_2 \in \|Q_2^t\|$ and $\varphi = \varphi_1 \vee \varphi_2$, we have

(1a) $\mathcal{T} \vdash w(F_1(\bar{c})) > 0$, *(1b)* $\mathcal{T} \vdash w(F_2(\bar{c})) > 0$,
(2a) $\mathcal{T} \vdash \varphi_1 \leftrightarrow F_1(\bar{c})$, *(2b)* $\mathcal{T} \vdash \varphi_2 \leftrightarrow F_2(\bar{c})$,

and there is no atom P in φ_1 or in φ_2 such that $\mathcal{T} \vdash P$ or $\mathcal{T} \vdash \neg P$. The last condition is obviously verified for the formula $\varphi_1 \vee \varphi_2$. Furthermore, by Lemma 1, from (1a) and (1b) it follows $\mathcal{T} \vdash w(F_1(\bar{c}) \vee F_2(\bar{c})) > 0$. Finally, by standard first-order reasoning, (2a) and (2b) are verified iff $\mathcal{T} \vdash \varphi_1 \vee \varphi_2 \leftrightarrow F_1(\bar{c}) \vee F_2(\bar{c})$ and we arrive at the result. □

For example, the above theorem is used for answering the query

$$\langle s/\text{stud}, p/\Lambda \mid w(\neg\text{takes}(s, \text{Algebra}) \vee \text{takes}(s, \text{Calculus})) = p\rangle$$

which asks for the tuples $\langle s, p\rangle$ such that p is the probability that if student s takes the Algebra course then it takes also the Calculus course.

The following two theorems enable to remove quantifiers in queries.

Theorem 29 *Let $\mathcal{T}$ be a probabilistic theory and $F(\bar{x}, y)$ a possibly quantified first-order formula with free variables among $\bar{x} = (x_1, \ldots, x_n)$ and y. Then*

(1) *If $|\,\theta\,|^t = \{\}$ then $\{\bar{x}/\bar{\tau} \mid (\forall y/\theta)F(\bar{x}, y)\}^t = |\,\bar{\tau}\,|^t$.*
(2) *If $|\,\theta\,|^t \neq \{\}$ then*

$$\{\bar{x}/\bar{\tau} \mid (\forall y/\theta)F(\bar{x}, y)\}^t = \{\bar{x}/\bar{\tau}, y/\theta \mid F(\bar{x}, y)\}^t \div |\,\theta\,|^t. \tag{8}$$

Proof. *Consider a query $Q^t = \langle \bar{x}/\bar{\tau} \mid (\forall y/\theta)F(\bar{x}, y)\rangle$ and its subquery $Q_1^t = \langle \bar{x}/\bar{\tau}, y/\theta \mid F(\bar{x}, y)\rangle$. We begin by proving (1). If $|\,\theta\,|^t = \{\}$, then θ's extension axiom in $\mathcal{T}$ is $(\forall x)\neg\theta(x)$ and thus $\mathcal{T} \vdash [(\forall y)\theta(y) \rightarrow F(\bar{c}, y))] \leftrightarrow$ true. Hence, $\bar{c}/$ true $\in \|\,Q^t\,\|$ iff $\bar{c}/$ true $\in |\,\bar{\tau}\,|^t$.*

For (2), suppose that θ's extension axiom in $\mathcal{T}$ is $(\forall x)(\theta(x) \leftrightarrow x = c^{(1)} \vee \ldots \vee x = c^{(r)})$. Then, $\mathcal{T} \vdash (\forall y/\theta)F(\bar{x}, y) \leftrightarrow \bigwedge_{i=1}^{r} F(\bar{c}, c^{(i)})$. By definition of division in t-relations, $\bar{c}/\varphi$ belongs to the right-hand side of (8) iff $\{\bar{c}c_1/\varphi_1, \ldots, \bar{c}c_r/\varphi_r\} \subseteq \|\,Q_1\,\|^t$ and $\varphi = \bigwedge_{i=1}^{r} \varphi_i$. By definition of answers to t-queries, we have for $i = 1, \ldots, r$, $\mathcal{T} \vdash \varphi_i \leftrightarrow F(\bar{c}, c_i)$, and $\mathcal{T} \vdash w(F(\bar{c}, c_i)) > 0$. Then, $\mathcal{T} \vdash (\bigwedge_{i=1}^{r} \varphi) \leftrightarrow (\bigwedge_{i=1}^{r} F(\bar{c}, c_i))$, and by Lemma 19, $\mathcal{T} \vdash w(\bigwedge_{i=1}^{r} F(\bar{c}, c_i)) > 0$. Thus, $\bar{c}/\varphi$ belongs to the left-hand side of (8). □

For example, the above theorem is used for answering the query

$$\langle c/\text{course}, p/\Lambda \mid w((\forall s/\text{stud})\neg\text{takes}(s, c)) = p\rangle$$

which asks for the tuples $\langle c, p\rangle$ such that p is the probability that course c is taken by no student.

Theorem 30 *Let $\mathcal{T}$ be a probabilistic theory and $F(\bar{x}, y)$ a possibly quantified first-order formula with free variables among $\bar{x} = (x_1, \ldots, x_n)$ and y. Then*

(1) *If $|\,\theta\,|^t = \{\}$ then $\{\bar{x}/\bar{\tau} \mid (\exists y/\theta)F(\bar{x}, y)\}^t = \{\}$.*
(2) *If $|\,\theta\,|^t \neq \{\}$ then*

$$\{\bar{x}/\bar{\tau} \mid (\exists y/\theta)F(\bar{x}, y)\}^t = \pi_{\bar{x},y}(\{\bar{x}/\bar{\tau}, y/\theta \mid F(\bar{x}, y)\}^t). \tag{9}$$

Proof. *Consider a query $Q^t = \langle \bar{x}/\bar{\tau} \mid (\exists y/\theta)F(\bar{x}, y)\rangle$ and its subquery $Q_1^t = \langle \bar{x}/\bar{\tau}, y/\theta \mid F(\bar{x}, y)\rangle$. We begin by proving (1). If $|\,\theta\,|^t = \{\}$ then θ's extension axiom in $\mathcal{T}$ is $(\forall x)\neg\theta(x)$ and thus $\mathcal{T} \vdash [(\exists y)\theta(y) \wedge F(\bar{c}, y))] \leftrightarrow$ false. Since by*

Lemma 1 $w(false) = 0$, *no* $\bar{c}/\varphi$ *satisfies the conditions for answers to t-queries and then* $\| Q^t \| = \{\}$.

For (2), suppose that θ*'s extension axiom in* $\mathcal{T}$ *is* $(\forall x)(\theta(x) \leftrightarrow x = c^{(1)} \vee \ldots \vee x = c^{(r)})$. *Then,* $\mathcal{T} \vdash (\exists y/\theta)F(\bar{x}, y) \leftrightarrow \bigvee_{i=1}^{r} F(\bar{c}, c^{(i)})$. *By definition of projection in t-relations,* $\bar{c}/\varphi$ *belongs to the right-hand side of (9) iff there is a* k *(*$1 \leq k \leq r$*) such that* $\{i_1, \ldots, i_r\}$ *is a permutation of* $\{1, \ldots, r\}$, $\{\bar{c}c_{i_1}/\varphi_{i_1}, \ldots, \bar{c}c_{i_k}\varphi_{i_k}\} \subseteq \| Q_1 \|^t$ *and* $\varphi = \bigvee_{j=1}^{k} \varphi_{i_j}$. *By definition of answers to t-queries, we have for* $j = 1, \ldots, k$, $\mathcal{T} \vdash \varphi_{i_j} \leftrightarrow F(\bar{c}, c_{i_j})$, *and by Lemma 1,* $\mathcal{T} \vdash w(F(\bar{c}, c_{i_j})) > 0$. *Then,* $\mathcal{T} \vdash (\bigvee_{i=1}^{k} \varphi_{i_j}) \leftrightarrow (\bigvee_{i=1}^{k} F(\bar{c}, c_{i_j}))$, $\mathcal{T} \vdash w(\bigvee_{i=1}^{k} F(\bar{c}, c_{i_j})) > 0$ *and thus,* $\bar{c}/\varphi$ *belongs to the left-hand side of (9).* □

For example, the above theorem is used for answering the query

$$\langle t/\text{prof}, s/\text{stud}, p/\Lambda \mid w((\exists c/\text{course})(\text{teaches}(t, c) \wedge \text{takes}(s, c))) = p \rangle$$

which asks for the tuples $\langle t, s, p \rangle$ such that p is the probability that student s takes at least one course c given by professor t.

Finally, the following theorem allow us to remove query variables which do not appear in the formula of the query. The easy proof is left to the reader.

Theorem 31 *Let* $\mathcal{T}$ *be a probabilistic theory and let* $F(\bar{x})$ *be a formula in which variable* y *does not occur free. Then*

(1) $\{y/\theta, \bar{x}/\bar{\tau} \mid F(\bar{x})\}^t = |\theta|^t \times \{\bar{x}/\bar{\tau} \mid F(\bar{x})\}^t$.
(2) *If for* $n \geq 1$, $\bar{x}/\bar{\tau} = x_1/\tau_1, \ldots, x_n/\tau_n$ *and for* $k \geq 0$, $\bar{z}/\bar{\phi} = z_1/\phi_1, \ldots, z_n/\phi_k$, *then*

$$\{\bar{x}/\bar{\tau}, y/\theta, \bar{z}/\bar{\phi} \mid F(\bar{x}, \bar{z})\}^t = \pi_{2,\ldots,n+1,1,n+2,\ldots,n+k+1}(|\theta|^t \times \{\bar{x}/\bar{\tau}, \bar{z}/\bar{\phi} \mid F(\bar{x}, \bar{z})\}^t).$$

Notice that the projection for case (2) above is needed only to permute the attributes of the answer in the right-hand side in the same order as the query variables in the left-hand side.

4.5 Evaluation of t-queries

As pointed out in Section 4, in order to evaluate first-order queries of the form $Q = \langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f \rangle$ where F is a first-order formula, we associate to Q a t-query $Q^t = \langle \bar{x}/\bar{\tau} \mid F(\bar{x}) \rangle$. The answer to such a t-query Q^t is composed of a set of tuples $\bar{c}/\varphi$ where φ is a propositional formula. This set of tuples can be seen as a t-relation.

All along the preceding sections we have studied the evaluation of t-queries. In this section we study how to obtain the answer to a first-order query Q from the answers to its associated t-query Q^t. Recall that the answer to a first-order query Q is a set of tuples $(\bar{c}, p)$ such that $\bar{c}$ satisfies the simple types $\bar{\tau}$, $p \in]0, 1]$, and $\mathcal{T} \vdash w(F(\bar{c})) = p$.

First, we show how to compute the probability of a formula φ from $\mathcal{F}_\mathcal{T}$ in a probabilistic theory $\mathcal{T}$. As in [34] , we first transform φ into a formula in disjunctive canonical form $\varphi' = D_1 \vee \ldots \vee D_n$ where each conjunct D_i contains every atom appearing in φ. Therefore we can obtain $w(\varphi) = w(\varphi') = w(D_1) + \cdots + w(D_n)$. This is shown in the next example.

Example 32 *Let $\varphi = A\overline{B} \vee \overline{A}C \vee \overline{B}C$ be a formula where $\{A, B, C\}$ are ground atoms, and suppose that, in a probabilistic theory $\mathcal{T}$, the probability of A is a, the probability of B is b, and so on, and let $\bar{p} = 1 - p$ for each probability p. The disjunctive canonical form is obtained by expanding φ as follows*

$$\begin{aligned}\varphi' &= A\overline{B}(C \vee \overline{C}) \vee \overline{A}(B \vee \overline{B})C \vee (A \vee \overline{A})\overline{B}C \\ &= A\overline{B}C \vee A\overline{B}\,\overline{C} \vee \overline{A}BC \vee \overline{A}\,\overline{B}C.\end{aligned}$$

Since every disjunct in φ' is mutually exclusive then

$$\begin{aligned}w(\varphi') &= w(A\overline{B}C) + w(A\overline{B}\,\overline{C}) + w(\overline{A}BC) + w(\overline{A}\,\overline{B}C) \\ &= a\bar{b}c + a\bar{b}\bar{c} + \bar{a}bc + \bar{a}\bar{b}c.\end{aligned}$$

An arbitrary trace formula φ involving n different atoms, can be interpreted as a Boolean function over n variables. Thus, φ can be transformed into disjunctive canonical form using a classical result in Boolean algebra (e.g. [20]). Indeed, every Boolean function $f(x_1, \ldots, x_n)$ can be expressed in the disjunctive canonical form by

$$f(x_1, \ldots, x_n) = \bigvee_{\bar{e}=\langle 0,\ldots,0\rangle}^{\bar{e}=\langle 1,\ldots,1\rangle} f(e_1, \ldots, e_n) x_1^{e_1} \cdots x_n^{e_n},$$

where $e_i = 0$ or 1, $x_j^0 = \bar{x_j}$, $x_j^1 = x_j$, $\bar{e} = \langle e_1, \ldots, e_n\rangle$ is an n-tuple of 0's and 1's, and the union extends over all 2^n combinations of n 0's and 1's for the e_i's.

Intuitively, the value of $f(e_1, \ldots, e_n)$ is equal to 0 or 1. If $f(e_1, \ldots, e_n) = 0$ the term $x_1^{e_1} \cdots x_n^{e_n}$ is absent (has a 0 multiplier) in the canonical form and if $f(e_1, \ldots, e_n) = 1$ the term $x_1^{e_1} \cdots x_n^{e_n}$ appears (has a 1 multiplier) in the canonical form.

Consider again formula $\varphi = AB \vee \overline{A}C \vee \overline{B}D \vee \overline{C}D$ of the previous example. Since there are 3 atoms, we evaluate φ for each of the eigth possible three-tuples $\langle e_1, e_2, e_3 \rangle$

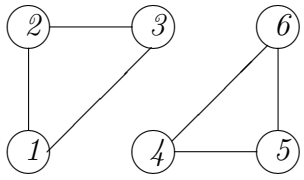
$$\varphi(0,0,1) = \varphi(0,1,1) = \varphi(1,0,0) = \varphi(1,0,1) = 1$$
$$\varphi(0,0,0) = \varphi(0,1,0) = \varphi(1,1,0) = \varphi(1,1,1) = 0.$$

Thus the disjunctive canonical form of φ has 4 terms $\overline{A}\,\overline{B}C \vee \overline{A}BC \vee A\overline{B}\,\overline{C} \vee A\overline{B}C$, as found in the previous example by expanding φ.

The next example shows how to split a formula φ into subformulas φ_i, such that each subformula can be evaluated independently.

Example 33 *Let $\{A, B, C, D, E, F, G, H\}$ be atomic formulas, consider $\varphi = AB \vee AC \vee BD \vee EFG \vee FH \vee GH$, and suppose that the probability of A is a, the probability of B is b, and so on. We draw a graph containing a node for each conjunct of φ and we establish the interrelations of conjuncts. This graph is constructed in two phases.*

First, two nodes are linked if they share a literal. For example, nodes 1 and 2 are linked since A appears in the first two conjuncts. The second phase consists in making the transitive closure of links, that is, if node i is linked to node j and if the latter is linked to node k, then nodes i and k are linked. This yields the following graph.



Since we obtain two disjoint subgraphs, the subformulas $\varphi_1 = AB \vee AC \vee BD$ and $\varphi_2 = EFG \vee FH \vee GH$ are independent. Therefore, each one of these subformulas can be independently evaluated. Thus, $w(\varphi_1) = ab + ac - abc + bd - abd$ and $w(\varphi_2) = efg - efgh + fh - fgh + gh$.

Since in probabilistic theories $\vdash w(\varphi_1 \vee \varphi_2) = 1 - w(\neg\varphi_1 \wedge \neg\varphi_2) = 1 - (w(\neg\varphi_1) \times w(\neg\varphi_2))$, then $w(\varphi) = 1 - (1 - w(\varphi_1))(1 - w(\varphi_2))$.

We now introduce a mapping *EVAL* which transforms a set of t-tuples $\bar{c}/\varphi$ into a set of tuples $(\bar{c}, p)$ given a probabilistic theory $\mathcal{T}$.

Definition 34 *Given a probabilistic theory $\mathcal{T}$ and a t-relation R, $EVAL(R) = S$ is given by*

$$S = \{(\bar{c}, p) \mid \bar{c}/\varphi \in R \wedge eval(\varphi) = p\}$$

where $eval(\varphi)$ is obtained by computing the probability of the disjunctive canonical form of φ as in the examples above.

We are now able to prove that the answer to a first-order query Q can be obtained applying the mapping $EVAL$ to the answer to its associated t-query Q^t. The following result is easily verified.

Theorem 35 *Let $\mathcal{T}$ be a probabilistic theory, $Q = \langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f \rangle$ a first-order query, and Q^t its associated t-query.*
Then $\| Q \| = EVAL(\| Q^t \|)$. $\square$

We next give some results about the complexity of evaluating first-order queries.

Definition 36 *For a trace formula φ, we define the length $|\varphi|$ as the number of distinct atoms appearing in φ.*

Let Q^t be a t-query, let E be an algebraic expression computing Q^t, let $\| E \|$ be the t-relation resulting from evaluating E over a given theory $\mathcal{T}$, and let $\text{card}(\| E \|)$ be the number of tuples in $\| E \|$. We establish the complexity of Q^t by giving an upper bound on the length of the trace formulas φ appearing in $\| E \|$.

Theorem 37 *For an algebraic expression E over t-relations, the upper bound $|E|$ on the length of the trace formulas in $\| E \|$ is computed inductively as follows:*

(1) *If $E \equiv R^t$ where R^t is a t-relation then $|E| = 1$.*
(2) *If $E \equiv \sigma(E_1)$ then $|E| = |E_1|$.*
(3) *If $E \equiv \pi(E_1)$ then $|E| = k\,|E_1|$ where $\text{card}(\| E_1 \|) = m$, $\text{card}(\| E \|) = n$ and $k = m - n + 1$.*
(4) *If $E \equiv E_1$ op E_2 then $|E| = |E_1| + |E_2|$, where op is one of $\cup$, $\cap$, $\times$, and $\bowtie$.*
(5) *If $E \equiv E_1 \div E_2$ then $|E| = k^2\,|E_1|\,|E_2|$, where $\text{card}(\| E_2 \|) = k$.*

Proof. *We consider only projection and division, since the other results follows from the definition of the operators.*

For projection, suppose that $\| E_1 \|$ has m tuples and that $\| E \|$ has n tuples where $m > n$. Intuitively this means that several tuples $\{\bar{c}\bar{d}_1/\varphi_1, \ldots, \bar{c}\bar{d}_j/\varphi_j\}$ of $\| E_1 \|$ are replaced by a tuple $\bar{c}/\varphi_1 \vee \ldots \vee \varphi_j$ in $\| E \|$. Therefore, at worst, the longest formula in $\| E \|$ will be $k = m - n + 1$ times the longest formula in $\| E_1 \|$.

For division, if $\| E_2 \|$ has k tuples, then the result follows since a tuple in $\| E \|$

is obtained by combining at most k tuples in $\| E_1 \|$ *with k tuples in* $\| E_2 \|$. $\square$

Notice that in our evaluation algorithm, we only divide a t-relation by a classical relation (corresponding to a simple type). In this particular case, result (5) above becomes

(5') If $E \equiv E_1 \div E_2$ then $|E| = k\,|E_1|$ where $\text{card}(\| E_2 \|) = k$.

We conclude by showing how the results of this section are used for recursively decompose queries during query evaluation. Consider the query

$$Q = \langle t/\text{prof}, s/\text{stud}, p/\Lambda \mid w((\exists c/\text{course})(\text{teaches}(t,c) \wedge \text{takes}(s,c))) = p \rangle$$

already given in Section 4.4. The answer $\| Q \|$ is computed as follows

$$\begin{array}{l}
EVAL(\{t/\text{prof}, s/\text{stud} \mid (\exists c/\text{course})(\text{teaches}(t,c) \wedge \text{takes}(s,c))\}^t) \\
EVAL(\pi_{1,2}\{t/\text{prof}, s/\text{stud}, c/\text{course} \mid \text{teaches}(t,c) \wedge \text{takes}(s,c)\}^t) \\
EVAL(\pi_{1,2}(\ \{t/\text{prof}, s/\text{stud}, c/\text{course} \mid \text{teaches}(t,c)\}^t \cap \\
\qquad\qquad \{t/\text{prof}, s/\text{stud}, c/\text{course} \mid \text{takes}(s,c)\}^t\ )) \\
EVAL(\pi_{1,2}(\ \pi_{2,1,3}(\ |\text{stud}|^t \times \{t/\text{prof}, c/\text{course} \mid \text{teaches}(t,c)\}^t\ ) \cap \\
\qquad\qquad (\ |\text{prof}|^t \times \{s/\text{stud}, c/\text{course} \mid \text{takes}(s,c)\}^t\ )\ )) \\
EVAL(\pi_{1,2}(\ \pi_{2,1,3}(\ |\text{stud}|^t \times (|\text{prof}|^t \times |\text{course}|^t) \cap |\text{teaches}|^t) \cap \\
\qquad\qquad (\ |\text{prof}|^t \times (|\text{stud}|^t \times |\text{course}|) \cap |\text{takes}|^t\ )\ )\ ))
\end{array}$$

Of course, multiple optimizations in the above decomposition are possible. However, the optimization of these algebraic expressions goes beyond the scope of this paper.

5 Probabilistic queries

So far, we have studied first-order queries of the form $\langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f \rangle$, where F is a first-order formula. We now study general queries of the form $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f) \rangle$, where F is an arbitrary formula. First, we need a definition.

Definition 38 *Consider a sequence* $\bar{\tau} = \tau_1, \ldots, \tau_n$ *of simple types. We associate to* $\bar{\tau}$ *a classical relation* $|\bar{\tau}|$ *projecting out the trace attribute from* $|\bar{\tau}|^t$.

Now, we make some minor restrictions in the form of probabilistic queries, restrictions which are motivated in the sequel.

Definition 39 *(Restriction to single-order formulas) In a probabilistic language $\mathcal{L}$, a formula F is said to be higher-order if F contains nested probability terms such as $w(w(P(x)) < w(Q(x))) = 0.7$. Similarly, a query Q is said to be higher-order if its formula is a higher-order formula.*

It is easy to verify that higher-order formulas of the form $w(w(F_1)\ \theta\ w(F_2))$, where θ is a comparison operator, always take either the value 1 or the value 0. Indeed, the inner term $w(F_1)\ \theta\ w(F_2)$ may be replaced either by *true* or by *false*, depending on whether the term is verifed or not. For this reason, we consider only single-order queries. This is not really a restriction since a higher-order query can be translated into an equivalent single-order one.

For example, let Q be the query $\langle x/\tau \mid w(w(P(x)) > w(R(x))) = c\rangle$. If $c = 1$, then Q is equivalent to $\langle x/\tau \mid w(P(x)) > w(R(x))\rangle$. If $c = 0$, then Q is equivalent to $\langle x/\tau \mid w(P(x)) \leq w(R(x))\rangle$. Otherwise, if $0 < c < 1$, then Q is equivalent to $\langle x/\tau \mid \mathit{false}\rangle$.

Definition 40 *(Evaluable queries) As it is well-known, not all queries in relational calculus can be answered sensibly when disjunction, negation, and universal quantification are allowed. The class of relational calculus queries or formulas that have sensible answers is called the domain independent class which is known to be undecidable. A large decidable subclass of domain independent formulas, called evaluable formulas, is defined in [37] . It comprises all other known subclasses of domain independent formulas such as range separable, range restricted, allowed or safe formulas. Further, the class of evaluable formulas is the largest decidable subclass of domain independent formulas that can be efficiently recognized.*

The class of evaluable queries is defined as follows.

Definition 41 *[37] Let F be a formula where*

$$dnf(F) = \%\bar{z}(D_1 \vee \ldots \vee D_n) \ \textit{ and } \ cnf(F) = \%\bar{z}(C_1 \wedge \ldots \wedge C_m)$$

are the conjunctive and disjunctive normal forms of F, and where % denotes a sequence of (possible mixed) quantifiers $\exists$ and $\forall$. Let θ be a comparison predicate and t a variable or a constant. Suppose also that F contains no negated comparison predicates, excepted $\neq$ (i.e. $\neg >$ is replaced by $\leq$). Then F is said to be evaluable iff the following properties hold:

(1) *For every free variable x in F, x occurs in a positive literal (other than $x = y$ or $x\ \theta\ t$) in every D_j.*

(2) *For every existentially quantifed variable x in F, x occurs in a positive literal (other than $x = y$ or $x\ \theta\ t$) in every D_j in which x occurs.*
(3) *For every universally quantifed variable x in F, x occurs in a negative literal (other than $x \neq y$) in every C_j in which x occurs.*

Notice that for a field variable y^f, saying that y^f appears in a positive (resp. negative) litteral in a formula F means that $w(G) = y^f$ (resp. $w(G) \neq y^f$) appears in F.

For example, the queries $Q_1 = \langle x/\tau, y^f/\Lambda \mid w(P(x)) = y^f \wedge y^f > 0.5\rangle$, $Q_2 = \langle x/\tau, y^f/\Lambda \mid w(P(x)) = y^f \vee w(Q(x)) = y^f\rangle$ are evaluable, whereas the queries $Q'_1 = \langle x/\tau, y^f/\Lambda, z^f/\Lambda \mid w(P(x)) = y^f \wedge z^f > y^f\rangle$, and $Q'_2 = \langle x/\tau, y^f/\Lambda, z^f/\Lambda \mid w(P(x)) = y^f \vee w(Q(x)) = z^f\rangle$, and $Q'_3 = \langle x/\tau, y^f/\Lambda \mid w(F(x,z)) = y^f \vee w(F(x,z)) \neq y^f\rangle$ are not evaluable.

Consider a probabilistic query $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f)\rangle$ where F is an arbitrary formula. Since the domain Λ is not finite, it is necessary that F be evaluable, otherwise $\|\,Q\,\|$ would not be finite. For example, query Q'_2 has an infinite number of answers since for each pair (c, p) such that $\mathcal{T} \vdash w(P(c)) = p$, there are infinitely many $q \in \Lambda$ such that $(c, p, q) \in \|\,Q'_2\,\|$.

Finally, note that for queries $Q = \langle \bar{x}/\bar{\tau} \mid F(\bar{x})\rangle$ having no free field variables, since in probabilistic theories every simple type is finite, then $\|\,Q\,\|$ has as most the same number of answers as in $|\,\bar{\tau}\,|$. Therefore $\|\,Q\,\|$ is finite, even if F is not evaluable.

In the following, we always suppose that queries are evaluable. Further we say that a formula F *instantiates the field variable* y^f if F is evaluable and y^f appears in F.

For example, the query $\langle x/\tau, y^f/\Lambda, z^f/\Lambda \mid w(F_1(x)) = y^f \wedge w(F_2(x)) = z^f\rangle$ instantiates variables y^f and z^f, whereas for the query $\langle x/\tau, y^f/\Lambda \mid w(F_1(x)) = y^f \wedge w(F_2(x)) > y^f\rangle$, if we define the subqueries $\langle x/\tau, y^f/\Lambda \mid w(F_1(x)) = y^f\rangle$ and $\langle x/\tau, y^f/\Lambda \mid w(F_2(x)) > y^f\rangle$, the latter subquery does not instantiate y^f.

5.1 Primitive probabilistic queries

We define a query as *primitive probabilistic* if it has one of the forms

$$\langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F_1(\bar{x})) = y^f\rangle,\ \langle \bar{x}/\bar{\tau} \mid w(F_1(\bar{x}))\ \theta\ c\rangle,\ \text{or}$$
$$\langle \bar{x}/\bar{\tau} \mid w(F_1(\bar{x}))\,\theta\,w(F_2(\bar{x}))\rangle$$

where F_1 and F_2 are first-order formulas and θ is a comparison predicate. The evaluation of the first type of queries has been studied in Section 4. The

following theorems state how to compute the answers for the other two types. Some easy proofs are left to the reader.

Theorem 42 *Let $\mathcal{T}$ be a probabilistic theory, let $F(\bar{x})$ be a first-order formula, let p be a real number belonging to $]0,1]$, and let θ be a comparison predicate. Then*

$$\{\bar{x}/\bar{\tau} \mid w(F(\bar{x}))\ \theta\ p\} \ = \ \pi_{\bar{x}}\sigma_{y^f\ \theta\ p}(\{\bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f\}).$$

The above theorem allows to evaluate queries such as

$$\langle s/\text{stud} \mid w((\forall c/\text{course})(\text{teaches}(\text{Anne}, c) \rightarrow \text{takes}(s, c))) > 0.5)\rangle$$

which asks for the students taking all courses given by Anne with a probability greater than 0.5.

Theorem 43 *Let $\mathcal{T}$ be a probabilistic theory, and let $F(\bar{x})$ be a first-order formula. Then*

$$\begin{array}{lll} (1)\ \{\bar{x}/\bar{\tau} \mid w(F(\bar{x})) > 0\} & = & \pi_{\bar{x}}(\{\bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f\}). \\ (2)\ \{\bar{x}/\bar{\tau} \mid w(F(\bar{x})) = 0\} & = & |\bar{\tau}| - \pi_{\bar{x}}(\{\bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F(\bar{x})) = y^f\}). \quad \square \end{array}$$

Intuitively, the query in (1) asks for tuples $\bar{c} \in \bar{\tau}$ satisfying F with a probability greater than 0. The answer to this query is obtained from the expression in the right-hand side by definition of query answers. Notice also that the query in (2) is equivalent to the query $\{\bar{x}/\bar{\tau} \mid \neg F(\bar{x})\}$ and is obtained by making the difference of $|\bar{\tau}|$ and the answer to the query in (1).

The above theorem allows to evaluate queries such as

$$\langle t/\text{prof} \mid w((\exists c/\text{course})(\text{course\_dep}(c, \text{`EE'}) \wedge \text{teaches}(t, c)) = 0)\rangle$$

which asks for the professors which for sure do not give at least one course in the 'EE' department.

Theorem 44 *Let $\mathcal{T}$ be a probabilistic theory, let $F_1(\bar{x})$ and $F_2(\bar{x})$ be first-order formulas and let $Q = \langle \bar{x}/\bar{\tau} \mid w(F_1(\bar{x})) = w(F_2(\bar{x}))\rangle$ be a query. If we define the subqueries $Q_1 = \langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F_1(\bar{x})) = y^f\rangle$ and $Q_2 = \langle \bar{x}/\bar{\tau}, z^f/\Lambda \mid w(F_2(\bar{x})) = z^f\rangle$, then*

$$\begin{aligned} \|\, Q \,\| = \pi_{\bar{x}}\sigma_{y^f = z^f}(\|\, Q_1 \,\| \bowtie_{\bar{x}} \|\, Q_2 \,\|) \cup \\ (\ |\bar{\tau}| - \pi_{\bar{x}}(\|\, Q_1 \,\|)\ ) \cap (\ |\bar{\tau}| - \pi_{\bar{x}}(\|\, Q_2 \,\|)\ ). \end{aligned} \tag{10}$$

Proof. *By definition of query answers, if p is a real number belonging to $]0,1]$ and if $\bar{c}$ is a tuple of object constants, then $\bar{c}$ belongs to the left-hand side of (10) iff (1) $\mathcal{T} \vdash \bar{\tau}(\bar{c})$; and (2) $\mathcal{T} \vdash w(F_1(\bar{c})) = w(F_2(\bar{c})) = p$. We have to distinguish two cases depending on whether (3) $\mathcal{T} \vdash p = 0$ or (4) $\mathcal{T} \vdash p \neq 0$.*

Let us analyze the first case. If p_1 and p_2 are real numbers belonging to $]0,1]$, then $(\bar{c}, p_1, p_2) \in (\| Q_1 \| \bowtie_{\bar{x}} \| Q_2 \|)$ iff

- $\mathcal{T} \vdash \bar{\tau}(\bar{c})$;
- $\mathcal{T} \vdash \Lambda(p_i) \neq 0$ *for* $i = 1, 2$;
- $\mathcal{T} \vdash p_i \neq 0$ *for* $i = 1, 2$; *and*
- $\mathcal{T} \vdash w(F_i(\bar{c})) = p_i$ *for* $i = 1, 2$.

Furthermore, $(\bar{c}, p_1, p_2) \in \sigma_{y^f = z^f}(\| Q_1 \| \bowtie \| Q_2 \|)$ iff in addition to the above conditions $\mathcal{T} \vdash p_1 = p_2$. By standard equality reasoning and by definition of classical projection, it follows that $\bar{c} \in \pi_{\bar{x}} \sigma_{y^f = z^f}(\| Q_1 \| \bowtie \| Q_2 \|)$ iff conditions (1)–(3) are verified.

For the second case, by Theorem 43, we have $\{\bar{x}/\bar{\tau} \mid w(F_i(\bar{x})) = 0\} = |\bar{\tau}| - \pi_{\bar{x}}(\| Q_i \|)$, for $i = 1, 2$. The result follows since $\bar{c} \in (\, |\bar{\tau}| - \pi_{\bar{x}}(\| Q_1 \|)\,) \cap (\, |\bar{\tau}| - \pi_{\bar{x}}(\| Q_2 \|)\,)$ iff conditions (1), (2), and (4) are verified. □

Intuitively, the above theorem states that the answer to $\| Q \|$ is composed of two parts. The first one is the set of tuples $\bar{c}$ such that $\langle \bar{c}, p \rangle \in \| Q_1 \|$ and $\langle \bar{c}, p \rangle \in \| Q_2 \|$ for a probability $p > 0$. The second part of the answer is the set of tuples $\bar{c}$ having probability 0 in both Q_1 and Q_2.

Theorem 45 *Let $\mathcal{T}$ be a probabilistic theory, let $F_1(\bar{x})$, $F_2(\bar{x})$ be first-order formulas, and let $Q = \langle \bar{x}/\bar{\tau} \mid w(F_1(\bar{x})) > w(F_2(\bar{x})) \rangle$ be a query. If we define the subqueries $Q_1 = \langle \bar{x}/\bar{\tau}, y^f/\Lambda \mid w(F_1(\bar{x})) = y^f \rangle$ and $Q_2 = \langle \bar{x}/\bar{\tau}, z^f/\Lambda \mid w(F_2(\bar{x})) = z^f \rangle$, then*

$$\| Q \| = \pi_{\bar{x}} \sigma_{y^f > z^f}(\| Q_1 \| \bowtie_{\bar{x}} \| Q_2 \|) \cup [\, \pi_{\bar{x}}(\| Q_1 \|) \cap (\, |\bar{\tau}| - \pi_{\bar{x}}(\| Q_2 \|)) \,].$$

Proof. *Similar to the proof of Theorem 44.* □

Intuitively, the above theorem states that the answer to $\| Q \|$ is composed of two parts. The first one is the set of tuples $\bar{c}$ such that $\langle \bar{c}, p \rangle \in \| Q_1 \|$ and $\langle \bar{c}, q \rangle \in \| Q_2 \|$ for probabilities $p, q > 0$ provided that $p > q$. The second part of the answer is the set of tuples $\bar{c}$ having probability greater than 0 in Q_1 and having probability 0 in Q_2.

The above theorem allows to evaluate queries such as

$$\langle c/\text{course} \mid w((\exists s/\text{stud})\text{takes}(s, c)) \; > \; w(\neg(\exists s/\text{stud})\text{takes}(s, c)) \rangle$$

which asks for the courses such that the probability that at least one student takes the course is greater than the probability that no student takes the course.

5.2 Compound probabilistic queries

Consider a compound probabilistic query $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f)\rangle$, where $\bar{y}^f = \langle y_1^f, \ldots, y_n^f\rangle$. For ease of evaluation, let F' be the formula obtained from F by rewriting every subformula of the form $w(G(\bar{x}))\ \theta\ y_i^f$, where θ is a comparison operator distinct from $=$, as $w(G(\bar{x})) = z_i^f \wedge z_i^f\ \theta\ y_i^f$, the z_i^f being variables not appearing in F. Furthermore, let $H(\bar{y}^f, \bar{z}^f)$ be the conjunction of all formulas $z_i^f\ \theta\ y_i^f$. It is easy to verify that

$$\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f)\} = \pi_{\bar{x},\bar{y}^f}\sigma_{H(\bar{y}^f,\bar{z}^f)}(\ \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda}, \bar{z}^f/\bar{\Lambda} \mid F'(\bar{x}, \bar{y}^f, \bar{z}^f)\}\ ).$$

Therefore, we suppose in the sequel that queries are rewritten in the above manner.

5.2.1 Conjunction

We first define a set of formulas $\mathcal{R}$ restricting the values that field variables can take. Since the domain Λ is not finite, we have to distinguish the case where a query have a subformula in $\mathcal{R}$.

Definition 46 *Given a probabilistic theory $\mathcal{T}$, we form the set of formulas $\mathcal{R}$ by starting with $y^f\ \theta\ c$ and $y^f\ \theta\ z^f$ where y^f, z^f are field variables and θ is a comparison predicate, and closing off under conjunction, disjunction, and negation.*

For example, $x^f > y^f \wedge y^f = z^f \vee x^f = 1.0$ is a formula in $\mathcal{R}$.

Consider a query $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \wedge F_2(\bar{x}, \bar{y}^f)\rangle$. We have to study different cases depending on the form of formulas F_1 and F_2. If both F_1 and F_2 instantiate every field variable from $\bar{y}^f$, the answer to Q is obtained from the intersection of the subqueries.

Theorem 47 *Let $\mathcal{T}$ be a probabilistic theory, and let F_1, F_2 be formulas where every field variable of $\bar{y}^f$ appears free and is instantiated in both F_1 and F_2. Then*

$$\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \wedge F_2(\bar{x}, \bar{y}^f)\} =$$

$$\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f)\} \cap \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_2(\bar{x}, \bar{y}^f)\}.$$

Proof. *Since every field variable of $\bar{y}^f$ appears free and is instantiated in both F_1 and F_2, the proof follows from the simple fact that, if $\bar{c}$, $\bar{p}$ are respectively tuples of object and field constants, then $\mathcal{T} \vdash F_1(\bar{c}, \bar{p}) \wedge F_2(\bar{c}, \bar{p})$ iff $\mathcal{T} \vdash F_1(\bar{c}, \bar{p})$ and $\mathcal{T} \vdash F_2(\bar{c}, \bar{p})$.* $\square$

The next example shows what happens if some field variables do not appear in both formulas F_1 or F_2.

Example 48 *Consider the query*

$$Q = \langle s/stud, p_1/\Lambda, p_2/\Lambda \mid w(takes(s, Algebra)) = p_1 \wedge w(takes(s, Calculus)) = p_2 \rangle$$

asking for the tuples $\langle s, p_1, p_2 \rangle$ such that p_1 is the probability that student s takes Algebra and p_2 is the probability that s takes Calculus. Let be the subqueries $Q_1 = \langle s/stud, p_1/\Lambda \mid w(takes(s, Algebra)) = p_1 \wedge \rangle$ and $Q_2 = \langle s/stud, p_2/\Lambda \mid w(takes(s, Calculus)) = p_2 \rangle$.

Suppose we have $\| Q_1 \| = \{\langle Peter, 1.0\rangle, \langle Paul, 0.8\rangle\}$ and $\| Q_2 \| = \{\langle Peter, 0.9\rangle, \langle Mary, 0.7\rangle\}$. Although Paul satisfies Q_1 with probability 0.8 he does not appears in $\| Q_2 \|$, i.e. he satifies Q_2 with probability 0. Since by the definition of query answers, $\langle Paul, 0.8, 0\rangle$ belongs to $\| Q \|$, we have to use the outer join to compute the answers to Q from $\| Q_1 \|$ and $\| Q_2 \|$.

We give in the sequel a definition of the outer join [14,4] . We slightly modify this definition in order to accommodate our purpose.

Definition 49 *Let r_1, r_2 be relations of scheme $\mathcal{R}_1(\bar{A}, \bar{B})$ and $\mathcal{R}_2(\bar{A}, \bar{C})$, where $\bar{A}$ is a tuple of object attributes and $\bar{B}, \bar{C}$ are tuples of field attributes. The outer join of r_1 and r_2 is given by:*

$$r_1 \overline{\bowtie} r_2 = r_1 \bowtie r_2 \cup \{(\bar{a}, \bar{b}, \bar{0}) \mid (\bar{a}, \bar{b}) \in r_1 \wedge \neg(\exists \bar{c})(\bar{a}, \bar{c}) \in r_2\} \cup \{(\bar{a}, \bar{0}, \bar{c}) \mid (\bar{a}, \bar{c}) \in r_2 \wedge \neg(\exists \bar{b})(\bar{a}, \bar{b}) \in r_1\}.$$

The outer join adds to $r_1 \bowtie r_2$ a set of tuples $(\bar{a}, \bar{b}, \bar{0})$ and $(\bar{a}, \bar{0}, \bar{c})$ for the tuples having the first attribute equal to $\bar{a}$ and appearing respectively only in r_1 or in r_2.

Consider again Example 48. By the above definition we have

$$\| Q_1 \| \overline{\bowtie} \| Q_2 \| = \{\langle \text{Peter}, 1.0, 0.9\rangle, \langle \text{Paul}, 0.8, 0\rangle, \langle \text{Mary}, 0, 0.7\rangle\}.$$

The next theorem states that $\|Q\| = \|Q_1\| \overline{\boxtimes} \|Q_2\|$.

Theorem 50 *Let $\mathcal{T}$ be a probabilistic theory, and let $F_1(\bar{x}, \bar{x}^f, \bar{y}^f)$ and $F_2(\bar{x}, \bar{y}^f, \bar{z}^f)$ be formulas instantiating all their field variables. Then*

$$\begin{aligned}\{\bar{x}/\bar{\tau}, \bar{x}^f/\bar{\Lambda}, \bar{y}^f/\bar{\Lambda}, \bar{z}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{x}^f, \bar{y}^f) \wedge F_2(\bar{x}, \bar{y}^f, \bar{z}^f)\} = \\ \{\bar{x}/\bar{\tau}, \bar{x}^f/\bar{\Lambda}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{x}^f, \bar{y}^f)\} \, \overline{\boxtimes}_{\bar{x}, \bar{y}^f} \\ \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda}, \bar{z}^f/\bar{\Lambda} \mid F_2(\bar{x}, \bar{y}^f, \bar{z}^f)\}. \qquad (11)\end{aligned}$$

Proof. *If $\bar{c}$ is a tuple of object constants and $\bar{p}_1$, $\bar{p}_2$, $\bar{p}_3$ are tuples of field constants, then $(\bar{c}, \bar{p}_1, \bar{p}_2, \bar{p}_3)$ belongs to the left-hand side of (11) iff*

- *either $\mathcal{T} \vdash F_1(\bar{c}, \bar{p}_1, \bar{p}_2) \wedge F_2(\bar{c}, \bar{p}_2, \bar{p}_3) \wedge (\bar{p}_1, \bar{p}_2) \neq \bar{0} \wedge (\bar{p}_2, \bar{p}_3) \neq \bar{0}$;*
- *either $\mathcal{T} \vdash F_1(\bar{c}, \bar{0}, \bar{0}) \wedge F_2(\bar{c}, \bar{p}_2, \bar{p}_3) \wedge (\bar{p}_2, \bar{p}_3) \neq \bar{0}$*
- *or $\mathcal{T} \vdash F_1(\bar{c}, \bar{p}_1, \bar{p}_2) \wedge F_2(\bar{c}, \bar{0}, \bar{0}) \wedge (\bar{p}_1, \bar{p}_2) \neq \bar{0}$.*

By definition of the outer join we arrive at the result. □

The next theorem allows to evaluate queries of the form $\langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \wedge F_2(\bar{x})\rangle$ where F_2 contains no field variables.

Theorem 51 *Let $\mathcal{T}$ be a probabilistic theory, let $F_1(\bar{x}, \bar{y}^f)$ and $F_2(\bar{x})$ be formulas such that F_1 instantiates every field variable from $\bar{y}^f$. Then*

$$\begin{aligned}\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \wedge F_2(\bar{x})\} = \\ \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f)\} \bowtie_{\bar{x}} \{\bar{x}/\bar{\tau} \mid F_2(\bar{x})\}\end{aligned}$$

Proof. *Follows from Theorem 42 and from the definition of query answers.* □

The above theorem allows to evaluate queries such as

$$\begin{aligned}\langle t/\text{prof} \,|\, w((\exists c/\text{course})(\text{course\_dep}(c, \text{`EE'}) \wedge \text{teaches}(t, c)) > 0.8 \wedge \\ (\exists c/\text{course})(\text{course\_dep}(c, \text{`CS'}) \wedge \text{teaches}(t, c))\rangle\end{aligned}$$

which asks for the professors which have a probability greater than 0.8 to give a course in the 'EE' department and which for sure give a course in the 'CS' department.

Finally, the last theorem allows to evaluate queries such as $\langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \wedge F_2(\bar{y}^f)\rangle$ where F_2 belongs to $\mathcal{R}$. The easy proof of the theorem

is omitted.

Theorem 52 *Let $\mathcal{T}$ be a probabilistic theory, let F_1 and F_2 be formulas such that F_1 instantiates every field variable from $\bar{y}^f$, $F_2 \in \mathcal{R}$, and where all the field variables of $\bar{y}^f$ may not appear in F_2. Then*

$$\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \wedge F_2(\bar{y}^f)\} = \sigma_{F_2(\bar{y}^f)}(\ \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f)\}\ ).$$

Given the query

$$\begin{aligned} Q = \langle s/\text{stud}, p_1/\Lambda, p_2/\Lambda \mid\ & w(\text{takes}(s, \text{Algebra})) = p_1 \wedge \\ & w(\text{takes}(s, \text{Calculus})) = p_2 \wedge p_1 > 0.8\rangle, \end{aligned}$$

the above theorem allows to compute $\| Q \|$ as follows

$$\begin{aligned} \| Q \| = \sigma_{p_1>0.8}(\{s/\text{stud}, p_1/\Lambda, p_2/\Lambda \mid\ & w(\text{takes}(s, \text{Algebra})) = p_1 \wedge \\ & w(\text{takes}(s, \text{Calculus})) = p_2\}\ ). \end{aligned}$$

5.2.2 Disjunction

Consider a query $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f)\rangle$. As already pointed out, the query formula $F(\bar{x}, \bar{y}^f)$ must be evaluable in order to obtain finitely many answers. In the case that F is of the form $F_1(\bar{x}, \bar{y}^f) \vee F_2(\bar{x}, \bar{y}^f)$, then F is evaluable if in particular both F_1 and F_2 instantiate every field variable y_i^f. The following theorem shows how to evaluate such queries.

Theorem 53 *Given a probabilistic theory $\mathcal{T}$, let F be a formula of the form $F_1(\bar{x}, \bar{y}^f) \vee F_2(\bar{x}, \bar{y}^f)$, both F_1 and F_2 instantiate every field variable y_i^f. Then*

$$\begin{aligned} &\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f) \vee F_2(\bar{x}, \bar{y}^f)\} = \\ &\qquad \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_1(\bar{x}, \bar{y}^f)\} \cup \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F_2(\bar{x}, \bar{y}^f)\}. \end{aligned}$$

Proof. *If F_1 and F_2 satisfy the conditions above, the proof follows from the simple fact that for tuples $\bar{c}$ of object constants and $\bar{p}$ of field constants, $\mathcal{T} \vdash F_1(\bar{c}, \bar{p}) \vee F_2(\bar{c}, \bar{p})$ iff $\mathcal{T} \vdash F_1(\bar{c}, \bar{p})$ or $\mathcal{T} \vdash F_2(\bar{c}, \bar{p})$.* □

5.2.3 Object Quantifiers

The following theorem allows to remove universal quantifiers in queries. However, we need to make a minor restriction. Consider the query $Q = \langle x/\tau, y^f/\Lambda \mid$

$(\forall z/\theta)(w(F(x,z)) = y^f)\rangle$. Recall that the query formula is an abbreviation of $(\forall z)(\theta(z) \rightarrow w(F(x,z)) = y^f)$ which is equivalent to $(\forall z)(\neg\theta(z) \vee w(F(x,z)) = y^f)$. If θ's extension axiom is $(\forall x)\neg\theta(x)$ then $\langle c, p\rangle \in \| Q \|$ for all $c \in \tau$ and $p \in \Lambda$. Therefore we disallow universal quantification over empty simple types.

Theorem 54 *Let $\mathcal{T}$ be a probabilistic theory and let $F(\bar{x}, \bar{y}^f, z)$ be a possibly quantified formula with free variables among $\bar{x}$, $\bar{y}^f$, and z. Suppose further that θ is a simple type whose extension axiom in $\mathcal{T}$ is not $(\forall x)\neg\theta(x)$. Then*

$$\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid (\forall z/\theta)F(\bar{x}, \bar{y}^f, z)\} = \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda}, z/\theta \mid F(\bar{x}, \bar{y}^f, z)\} \div |\theta| \,. \tag{12}$$

Proof. *Suppose that θ's extension axiom in $\mathcal{T}$ is as follows:*

$$(\forall x)(\theta(x) \leftrightarrow x = a_1 \vee \ldots \vee x = a_r).$$

Let $\bar{c}$ and $\bar{p}$ be, respectively, tuples of object and field constants. Then

$$\begin{array}{l}
\mathcal{T} \vdash (\forall z)(\theta(z) \rightarrow F(\bar{c}, \bar{p}, z)) \quad \textit{iff} \\
\mathcal{T} \vdash (\forall z)((z = a_1 \vee \ldots \vee z = a_r) \rightarrow F(\bar{c}, \bar{p}, z)) \quad \textit{iff} \\
\mathcal{T} \vdash (\forall z)(z = a_i \rightarrow F(\bar{c}, \bar{p}, z)), \quad \textit{for } i = 1, \ldots, r \quad \textit{iff} \\
\mathcal{T} \vdash F(\bar{c}, \bar{p}, a_i), \quad \textit{for } i = 1, \ldots, r \quad \textit{iff} \\
\mathcal{T} \vdash F(\bar{c}, \bar{p}, a) \textit{ for every } a \in |\theta|.
\end{array}$$

Hence, a tuple $(\bar{c}, \bar{p})$ is an element of the left-hand side of (12) iff

$$\begin{array}{ll}
\mathcal{T} \vdash \bar{\tau}(\bar{c}); & (13) \\
\mathcal{T} \vdash \bar{\Lambda}(\bar{p}); & (14) \\
\mathcal{T} \vdash p_i \neq 0 \ \textit{ for at least one } i = 1, \ldots, n; \quad \textit{and} & (15) \\
\mathcal{T} \vdash (\forall z)(\theta(z) \rightarrow F(\bar{c}, \bar{p}, z)). &
\end{array}$$

By the preamble of this proof, the last formula is equivalent to

$$\mathcal{T} \vdash F(\bar{c}, \bar{p}, a) \textit{ for every } a \in |\theta| \,. \tag{16}$$

Since by Lemma 17, $a \in |\theta|$ iff $\mathcal{T} \vdash \theta(a)$, formulas (13)– (16) are verified iff for every $a \in |\theta|$, $(\bar{c}, \bar{p}, a) \in \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda}, z/\theta \mid F(\bar{x}, \bar{y}^f, z)\}$, i.e., iff $(\bar{c}, \bar{p})$ is an element of the right-hand side of (12). □

The above theorem allows to evaluate queries such as

$$\langle t/\text{prof} \mid (\forall c/\text{course})(w(\text{teaches}(t,c)) > w((\forall s/\text{stud})\neg\text{takes}(s,c)))\rangle$$

which asks for the professors t such that for all courses c the probability that t teaches c is greater than the probability that no student takes c.

The following theorem allows to remove existential quantifiers over object variables.

Theorem 55 *Let $\mathcal{T}$ be a probabilistic theory and let $F(\bar{x}, \bar{y}^f, z)$ be a possibly quantified formula with free variables among $\bar{x}$, $\bar{y}^f$, and z. Then*

(1) *If $|\theta| = \{\}$ then $\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid (\exists z/\theta)F(\bar{x}, \bar{y}^f, z)\}^t = \{\}$.*
(2) *If $|\theta| \neq \{\}$ then*

$$\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid (\exists y/\theta)F(\bar{x}, \bar{y}^f, z)\} = \pi_{\bar{x},\bar{y}^f}\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda}, z/\theta \mid F(\bar{x}, \bar{y}^f, z)\}. \quad (17)$$

Proof. *Result (1) is trivial. For (2), suppose that θ's extension axiom in $\mathcal{T}$ is as follows:*

$$(\forall x)(\theta(x) \leftrightarrow x = a_1 \vee \ldots \vee x = a_r).$$

Let $\bar{c}$ and $\bar{p}$ be, respectively, tuples of object and field constants. Then

$$\begin{array}{l}
\mathcal{T} \vdash (\exists z)(\theta(z) \wedge F(\bar{c}, \bar{p}, z)) \quad \textit{iff} \\
\mathcal{T} \vdash (\exists z)((z = a_1 \vee \ldots \vee z = a_r) \wedge F(\bar{c}, \bar{p}, z)) \quad \textit{iff} \\
\mathcal{T} \vdash (\exists z)(\bigvee_{i=1}^r z = a_i \wedge F(\bar{c}, \bar{p}, z)) \quad \textit{iff} \\
\mathcal{T} \vdash \bigvee_{i=1}^r F(\bar{c}, \bar{p}, a_i) \quad \textit{iff} \\
\mathcal{T} \vdash F(\bar{c}, \bar{p}, a) \textit{ for an } a \in |\theta|.
\end{array}$$

A tuple $(\bar{c}, \bar{p})$ is an element of the left-hand side of (17) iff

$$\begin{array}{ll}
\mathcal{T} \vdash \bar{\tau}(\bar{c}); & (18) \\
\mathcal{T} \vdash \bar{\Lambda}(\bar{p}); & (19) \\
\mathcal{T} \vdash p_i \neq 0 \textit{ for at least one } i = 1, \ldots, n; \quad \textit{and} & (20) \\
\mathcal{T} \vdash (\exists z)(\theta(z) \wedge F(\bar{c}, \bar{p}, z)) &
\end{array}$$

By the preamble of this proof, the last formula is equivalent to

$$\mathcal{T} \vdash F(\bar{c}, a) \textit{ for an } a \in |\theta|. \quad (21)$$

Since, by Lemma 17, $a \in |\theta|$ iff $\mathcal{T} \vdash \theta(a)$, formulas (18)– (21) hold iff for an $a \in |\theta|$, $(\bar{c}, \bar{p}, a) \in \{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda}, z/\theta \mid F(\bar{x}, \bar{y}^f, z)\}$, i.e., iff $(\bar{c}, \bar{p})$ is an element of the right-hand side of (17). □

The above theorem allows to evaluate queries such as

$$\langle t/\text{prof} \mid (\exists c/\text{course}) w(\neg\text{teaches}(t,c)) > 0.8)\rangle$$

which asks for the professors having a probability greater than 0.8 to do not give a course.

5.2.4 Field quantifiers

In this section we study the evaluation of queries having quantifiers over field variables. First of all notice that some of the existential field quantifiers (but not all) can be removed. For example, the queries

$$Q_1 = \langle x/\tau \mid (\exists y^f/\Lambda)(w(F(x)) = y^f)\rangle,$$
$$Q_2 = \langle x/\tau \mid (\exists y^f/\Lambda)(w(F(x)) = y^f \wedge y^f > c)\rangle, \text{ and}$$
$$Q_3 = \langle x/\tau \mid (\exists y^f/\Lambda)(w(F_1(x)) = y^f \wedge w(F_2(x)) \;\theta\; y^f)\rangle$$

are respectively equivalent to the queries $Q'_1 = \langle x/\tau \mid \mathit{true}\rangle$, $Q'_2 = \langle x/\tau \mid w(F(x)) > c\rangle$ and $Q'_3 = \langle x/\tau \mid w(F_1(x)) \;\theta\; w(F_2(x))\rangle$. However, the quantifier cannot be removed in

$$Q_4 = \langle x/\tau \mid (\exists y^f/\Lambda)(\forall z/\theta)(w(F_1(x,z)) = y^f \rightarrow w(F_2(x)) < y^f)\rangle.$$

With respect to universal field quantifiers, all of them can be replaced by existential field quantifiers. For example, query $Q = \langle x/\tau \mid (\forall y^f/\Lambda)(w(F_1(x)) = y^f \rightarrow w(F_2(x)) = y^f)\rangle$ is equivalent to $Q = \langle x/\tau \mid (\exists y^f/\Lambda)(w(F_1(x)) = y^f \wedge w(F_2(x)) = y^f)\rangle$ and finally is equivalent to $Q'' = \langle x/\tau \mid w(F_1(x)) = w(F_2(x))\rangle$.

Without loss of generality, consider an evaluable query in CNF

$$Q \equiv \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid (\forall z^f/\Lambda)(\%\bar{w}/\bar{\theta})(C_1 \wedge \ldots \wedge C_m)\rangle$$

where % denotes a sequence of (possible mixed) quantifiers $\exists$ and $\forall$. By definition of evaluable queries, z^f occurs in a negative literal (other than $z^f \neq z'^f$) in every C_j in which z^f occurs. Therefore, every C_j in which z^f occurs is of the following form $C_j \equiv w(F_j) = z^f \rightarrow G_j$. Defining $C'_j \equiv C_j$ if z^f does not occur in C_j, and $C'_j \equiv w(F_j) = z^f \wedge G_j$ if z^f occurs in C_j, then Q is equivalent to the query

$$Q' \equiv \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid (\exists z^f/\Lambda)(\%\bar{w}/\bar{\theta})(C'_1 \wedge \ldots \wedge C'_m)\rangle.$$

The next theorem states how to eliminate existential field quantifiers.

Theorem 56 *Let $\mathcal{T}$ be a probabilistic theory and let $F(\bar{x}, \bar{y}^f, z)$ be a possibly quantified formula with free variables among $\bar{x}$, $\bar{y}^f$, and z. Then*

$$\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid (\exists z^f/\Lambda)F(\bar{x}, \bar{y}^f, z^f)\} = \\ \pi_{\bar{x},\bar{y}^f}\{\bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda}, z^f/\Lambda \mid F(\bar{x}, \bar{y}^f, z^f)\}.$$

Proof. *Similar to the proof of Theorem 55.* □

The above theorem allows to evaluate queries such as

$$\langle t/\text{prof}, c/\text{course} \mid (\exists p/\Lambda)(w(\neg\text{teaches}(t, c)) = p \wedge \\ (\forall s/\text{stud})w(\text{takes}(s, c)) > p)\rangle$$

which asks for the couples $\langle t, c\rangle$ such that if the probability that professor t does not teach course c is p then, for each student s, the probability that s takes course c is greater than p.

Finally, the last theorem allows to eliminate query variables that do not appear in the query formula. The easy proof is left to the reader.

Theorem 57 *Let $\mathcal{T}$ be a probabilistic theory, let $\bar{x}/\bar{\phi} = \langle x_1/\phi_1, \ldots, x_n/\phi_n\rangle$ be a tuple of object and field variables, and let $F(\bar{x})$ be a formula in which the object variable y is not free. Then*

(1) $\{y/\theta, \bar{x}/\bar{\phi}, \mid F(\bar{x})\} = \mid\theta\mid \times \{\bar{x}/\bar{\phi} \mid F(\bar{x})\}$.
(2) *If for $k \geq 0$, $\bar{z}/\bar{\psi} = \langle z_1/\psi_1, \ldots, z_k/\psi_k\rangle$ then*

$$\{\bar{x}/\bar{\phi}, y/\theta, \bar{z}/\bar{\psi} \mid F(\bar{x}, \bar{z}))\} = \\ \pi_{2,\ldots,n+1,1,n+2,\ldots,n+k+1}(\ \mid\theta\mid \times \{\bar{x}/\bar{\phi}, \bar{z}/\bar{\psi} \mid F(\bar{x}, \bar{z})\}\ ).$$

6 Related work

The need for uncertainty management in database and knowledge-base systems has motivated much of the work on the logical foundations of reasoning with uncertain knowledge. In this context, probability theory is the most widely accepted formalism for reasoning about change and uncertainty. We review in this section related approaches concerning probabilistic extensions of (deductive) databases and logic programming.

We described in this paper an extension of the relational model allowing to capture a particular type of probabilistic information. In order to formalize

probabilistic relational databases and to study query evalation, we needed a logic for reasoning about probability. Although there is a wealth of literature available on probabilistic logic (see for example the references in [5]), the foundations of our work was given by Halpern, which studied in [9] several first-order logics of probabilities. He considered two approaches to giving semantics to such logics. The first approach puts a probability on the domain, and is appropriate for giving semantics to formulas involving statistical information such as "the probability that a randomly chosen student lives in Brussels is greater than 0.9". The second approach puts a probability on possible worlds and is appropriate for giving semantics to formulas describing degrees of belief such as "the probability that Peter (a particular student) lives in Brussels is greater than 0.9". It is this logic that we used for formalizing probabilistic relational databases. In addition, Halpern showed that both approaches can be easily combined, allowing to reason about statistical information and degrees of belief. Halpern also gave axiom systems that are sound and complete in cases where a complete axiomatization is possible.

In the context of logic programming, the introduction of probability has been studied by Ng and Subrahmanian in [23–25] . They defined a logical framework where conjunctions and disjunctions are annotated with closed intervals of truth values $[\rho_1, \rho_2]$ where ρ_i may contain constants, variables or interpreted functions. They developed fixpoint and model-theoretic semantics and provided a sound and (weakly) complete proof procedure. As explained in the next section, several of their results can be used when extending our framework. Notice that we allow general queries of the form $Q = \langle \bar{x}/\bar{\tau}, \bar{y}^f/\bar{\Lambda} \mid F(\bar{x}, \bar{y}^f)\rangle$ for any well-formed formula F, in particular allowing negation and universal quantification over both field and object variables. In [23–25] universal quantifiers are not allowed in queries.

In [8,13] , Kießling et. al. studied the problem of reasoning in the presence of incomplete information and proposed a sound (propositional) probabilistic calculus based on conditional probabilities. However, this approach is less general than the work of Ng and Subrahmanian.

One criticism leveled against probabilistic approaches for uncertainty management is how the probabilities representing degree of likelihood can be derived. Lakshmanan in [15] observed that beliefs (and doubts) are formed by agents using underlying scenarios in the context of which the facts or rules are believed (or doubted). Thus, he proposed a framework in which the facts and rules of a knowledge-base are associated with propositional formulas representing the scenarios where a fact/rule is believed and doubted. Computation of probabilities is accomplished by compiling the belief and doubt information into a linear program deriving bounds on belief and doubt probabilities. This technique is related to our evaluation of t-relations and can be used in our approach if we drop the independence assumptions in probabilistic theories.

Also, Lakshmanan and Sadri proposed an approach to probabilistic deductive databases [17] based on a tri-lattice of probabilistic truth values. Using their framework, it is possible to reason with facts and rules having associated ranges of probabilities indicating belief and doubt.

In a conceptually different approach, Sadri [34] studies how to calculate reliability of answers to a query in a relational database where information comes from sources of different reliabilities. That approach allows for the representation of the contributing sources of each piece of information in a database by associating to each tuple a vector of length k with -1, 0 and 1 entries, where k is the number of information sources. To a k-vector corresponds a propositional expression specifying the condition under which the tuple exists in terms of the propositional variables representing information sources. Thus, these extended relations are similar to our t-relations, and indeed the extended algebraic operators defined in [34] are similar to ours, except for division operator which is not defined there. That framework was extended to deductive databases in [16] . Both works make the assumption of independence between information contributed by different sources. Similarly, we have assumed independence of events in our framework.

7 Summary and conclusions

Information of a stochastic nature is very common in real-life situations. We have shown that two different types of probabilistic information can be introduced into a relational database. We have then focused on manipulating one of these types and defined probabilistic relations.

Probabilistic databases are formalized using a probabilistic logic language proposed by Halpern. That logic is a suitable formalism for representing probabilistic information, as well as for precisely stating the semantics assigned to probabilistic databases. We represented probabilistic databases by means of probabilistic theories and studied query evaluation.

We distinguished two types of queries: first-order and probabilistic queries. For the evaluation of the former, we introduced a special type of relations, called trace relations or t-relations, allowing to manipulate probabilistic information by keeping track of the origin of tuples. We also generalized the relational operators for t-relations.

As we have shown, the evaluation of first-order queries can be obtained by manipulating t-relations. In this way, given a first-order query Q, we evaluate an associated t-query Q^t which gives a t-relation as result. The answer to the original query is then obtained with a mapping $EVAL$ which, based on the

assertions of the probabilistic theory, evaluates the t-relation $\| Q^t \|$ and gives as result a probabilistic relation $\| Q \|$. Finally, we studied the evaluation of probabilistic queries. The evaluation of such a query Q is obtained by applying the classical relational operators to the subqueries composing Q.

Our work can be extended in two directions. The first allows the probability of events to be closed intervals. We can use the results of [23] , in particular the two operators $\otimes$ and $\oplus$ for combining intervals.

The probabilistic theories studied in this paper contain a set of axioms stating that all the events represented in the database are independent. The second extension relaxes this restriction in order to accomodate real-life situations. This amounts to allow the indepence axioms in probabilistic theories to be arbitrary field formulas. Several results developed in the related works reviewed in Section 6 can be used for query evaluation in probabilistic theories having arbitrary field formulas. Notice that t-relations are extremely important in this context because, as stated in [33] , they allow to defer the evaluation of probabilities to the last stage where all the relational operators have already been computed. Thus, for the evaluation of queries when general independence axioms are allowed, it suffices to generalize the *EVAL* mapping by capturing the constraints on the probabilities in the form of a linear program as done in [15] .

Introducing probabilistic information into existing relational database management systems requires to be able to manipulate t-relations. Since t-relations are classical relations extended with an additional column containing propositional formulas, the relational database management systems have to be extended with a component for manipulating propositional formulas. Since the manipulation of propositional formulas is a well-studied problem (e.g. in the theory of switching circuits), this component is easy to realize.

Acknowledgements

I woulk like to thank professor Alain Pirotte, my thesis advisor, for his many helpful comments and suggestions. Many thanks also to professor Philippe Smets for helpful discussions. Finally, I am grateful to the referees whose careful reading helped to considerably improve this paper.

References

[1] D. Barbará, H. García-Molina, and D. Porter. A probabilistic relational data model. In F. Bancilhon, C. Thanos, and D. Tsichritzis, editors, *Proc. of the Int.*

Conf. on Extending Database Technology, EDBT'90, LNCS 416, pages 60–74, Venice, Italy, 1990. Springer-Verlag.

[2] R. Cavallo and M. Pittarelli. The theory of probabilistic databases. In *Proc. 13th Int. Conf. on Very Large Databases*, Brighton, U.K., 1987.

[3] E. Codd. Extending the database relational model to capture more meaning. *ACM Trans. on Database Systems*, 4(4):397–434, Dec. 1979.

[4] R. Elmasri and S. Navathe. *Fundamentals of Database Systems*. Benjamin/Cummings, 2 edition, 1994.

[5] R. Fagin, J. Halpern, and N. Meggido. A logic for reasoning about probabilities. *Information and Computation*, 87:78–128, 1990.

[6] E. Gelenbe and G. Hebrail. A probability model of uncertainty in databases. In *Proc. of the Int. Conf. on Data Engineering*, 1986.

[7] G. Grahne. *The Problem of Incomplete Information in Relational Databases*. LNCS 554. Springer-Verlag, 1991.

[8] U. Güntzer, W. Kießling, and H. Thöne. New directions for uncertainty reasoning in deductive databases. In *Proc. ACM-SIGMOD Int. Conf. on Management of Data*, pages 178–187, Denver, 1991.

[9] J. Halpern. An analysis of first–order logics of probability. *Artificial Intelligence*, 46(3):311–350, June 1990.

[10] G. Hulin, A. Pirotte, D. Roelants, and M. Vauclair. Logic and databases. In A. Thayse, editor, *From Modal Logic to Deductive Databases*, pages 279–350. Wiley, 1989.

[11] T. Imieliński and W. Lipski, Jr. Incomplete information in relational databases. *Journal of the ACM*, 31(4):761–791, Oct. 1984.

[12] T. Imielinski and K. Vadaparty. Complexity of query processing in databases with or-objects. In *Proc. 8th ACM SIGACT-SIGMOD Symp. on Principles of Database Systems*, 1989.

[13] W. Kießling, H. Thöne, and U. Güntzer. Database support for problematic knowledge. In A. Pirotte, C. Delobel, and G. Gottlob, editors, *Proc. of the Int. Conf. on Extending Database Technology, EDBT'92*, Vienna, Austria, 1992. Springer-Verlag.

[14] M. Lacroix and A. Pirotte. Generalized joins. *ACM SIGMOD Record*, 8(3), Sept. 1976.

[15] V. Lakshmanan. An epistemic foundation for logic programming with uncertainty. In *Proc. of the Int. Conf. on Foundations of Software Technology and Theoretical Computer Science*, LNCS 880, pages 197–207, Madras, India, Dec. 1994. Springer-Verlag.

[16] V. Lakshmanan and F. Sadri. Modeling uncertainty in deductive databases. In *Proc. of the Int. Conf. on Database Expert Systems and Applications, DEXA'94*, LNCS 856, pages 197–207, Athens, Greece, Sept. 1994. Springer-Verlag.

[17] V. Lakshmanan and F. Sadri. Probabilistic deductive databases. In *Proc. of the Int. Logic Programming Symposium*, pages 197–207, Ithaca, NY, Nov. 1994. MIT Press.

[18] K. Liu and R. Sunderraman. Indefinite and maybe information in relational databases. *ACM Trans. on Database Systems*, 15(1):1–39, Mar. 1990.

[19] K. Liu and R. Sunderraman. A generalized relational model for indefinite and maybe information. *IEEE Trans. on Knowledge and Data Engineering*, 3(1):65–77, Mar. 1991.

[20] R. Miller. *Switching Theory*, volume 1: Combinatorial Circuits. John Wiley & Sons, Inc., 1965.

[21] J. Minker. On indefinite databases and the closed world assumption. In D. W. Loveland, editor, *Proceedings of the 6th Conference on Automated Deduction*, LNCS 138, pages 292–308, New York, USA, June 1982. Springer-Verlag.

[22] J. Minker. Toward a foundation of disjunctive logic programming. In E. Lusk and R. Overbeek, editors, *Proceedings of the North American Logic Programming Conference*, pages 1215–1235. MIT Press, 1989.

[23] R. Ng and V. Subrahmanian. Probabilistic logic programming. *Information and Computation*, 101(2):150–201, 1992.

[24] R. Ng and V. Subrahmanian. A semantical framework for supporting subjective and conditional probabilities in deductive databases. *Journal of Automated Reasoning*, 10(2):191–235, 1993.

[25] R. Ng and V. Subrahmanian. Stable semantics for probabilistic databases. *Information and Computation*, 110(1):42–83, 1994.

[26] M. Pittarelli. An algebra for probabilistic databases. *IEEE Trans. on Knowledge and Data Engineering*, 6(2):293–303, 1994.

[27] A. Rajasekar, J. Lobo, and J. Minker. Weak generalized closed world assumption. *Journal of Automated Reasoning*, 5(3):293–307, 1989.

[28] K. Raju and A. K. Majumdar. Fuzzy functional dependencies and lossless join decomposition of fuzzy relational database systems. *ACM Trans. on Database Systems*, 13(2):129–166, 1988.

[29] R. Reiter. Towards a logical reconstruction of relational database theory. In M. Brodie, J. Mylopoulos, and J. Schmidt, editors, *On Conceptual Modelling*, pages 191–238. Springer-Verlag, Berlin, 1984.

[30] R. Reiter. A sound and sometimes complete query evaluation algorithm for relational databases with null values. *Journal of the ACM*, 33(2):349–370, Apr. 1986.

[31] R. Reiter and J. de Kleer. Foundations of assumption-based truth maintenance system: Preliminary report. In *Proc. of the AAAI-87*, pages 183–188, 1987.

[32] K. Ross and R. Topor. Inferring negative information from disjunctive databases. *Journal of Automated Reasoning*, 4:397–424, 1988.

[33] F. Sadri. Modeling uncertainty in databases. In *Proc. of the 7th IEEE Int. Conf. on Data Engineering*, pages 122–131, 1991.

[34] F. Sadri. Reliability of answers to queries in relational databases. *IEEE Trans. on Knowledge and Data Engineering*, 3(2):245–251, 1991.

[35] J. Shoenfield. *Mathematical Logic*. Addison-Wesley, Massachusetts, 1967.

[36] L. Sombe. *Reasoning under Uncertain Information in Artificial Intelligence*. Wiley, 1990.

[37] A. Van Gelder and R. Topor. Safety and translation of relational calculus queries. *ACM Trans. on Database Systems*, 16(2):235–278, 1991.

[38] L. Yuan and D.-A. Chiang. A sound and complete query evaluation algorithm for relational databases with null values. In *Proc. ACM-SIGMOD Int. Conf. on Management of Data*, pages 74–81, Chicago, June 1988.

[39] L. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[40] L. Zadeh. Fuzzy sets as a basis for theory of possibility. *Fuzzy Sets Systems*, 1(1):3–28, 1978.

[41] L. Zadeh. A thery of approximate reasoning. In J. H. et al., editor, *Machine Intelligence 9*, pages 149–194. Ellis Hoorwood Ltd., Sussex, UK, 1985.

[42] E. Zimányi. *Incomplete and Uncertain Information in Relational Databases*. PhD thesis, Université Libre de Bruxelles, Belgium, July 1992.

[43] E. Zimányi and A. Pirotte. Imperfect knowledge in databases. In A. Motro and P. Smets, editors, *Uncertainty Management in Information Systems: from Needs to Solutions*. Kluwer, 1996. In press. Long version in Research Report RR 92-36, Unité d'Informatique, Faculté des Sciences Appliquées, UCL.