I N F S Y S

R E S E A R C H

R E P O R T

INSTITUT FÜR INFORMATIONSSYSTEME

ABTEILUNG WISSENSBASIERTE SYSTEME

P-$\mathcal{SHOQ}(\mathbf{D})$: A PROBABILISTIC EXTENSION OF $\mathcal{SHOQ}(\mathbf{D})$ FOR PROBABILISTIC ONTOLOGIES IN THE SEMANTIC WEB

Rosalba Giugno and Thomas Lukasiewicz

INFSYS RESEARCH REPORT 1843-02-06

APRIL 2002

Institut für Informationssysteme
Abtg. Wissensbasierte Systeme
Technische Universität Wien
Favoritenstraße 9-11
A-1040 Wien, Austria
Tel: +43-1-58801-18405
Fax: +43-1-58801-18493
sek@kr.tuwien.ac.at
www.kr.tuwien.ac.at

TECHNISCHE UNIVERSITÄT WIEN

INFSYS RESEARCH REPORT
INFSYS RESEARCH REPORT 1843-02-06, APRIL 2002

P-$\mathcal{SHOQ}(\mathbf{D})$: A PROBABILISTIC EXTENSION OF $\mathcal{SHOQ}(\mathbf{D})$ FOR PROBABILISTIC ONTOLOGIES IN THE SEMANTIC WEB

(PRELIMINARY REPORT, 28 JUNE 2002)

Rosalba Giugno[1] and Thomas Lukasiewicz[2]

Abstract. Ontologies play a central role in the development of the semantic web, as they provide precise definitions of shared terms in web resources. One important web ontology language is DAML+OIL; it has a formal semantics and a reasoning support through a mapping to the expressive description logic $\mathcal{SHOQ}(\mathbf{D})$ with the addition of inverse roles. In this paper, we present a probabilistic extension of $\mathcal{SHOQ}(\mathbf{D})$, called P-$\mathcal{SHOQ}(\mathbf{D})$, to allow for dealing with probabilistic ontologies in the semantic web. The description logic P-$\mathcal{SHOQ}(\mathbf{D})$ is based on the notion of probabilistic lexicographic entailment from probabilistic default reasoning. It allows to express rich probabilistic knowledge about concepts and instances, as well as default knowledge about concepts. We also present sound and complete reasoning techniques for P-$\mathcal{SHOQ}(\mathbf{D})$, which are based on reductions to classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$ and to linear programming, and which show in particular that reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$ is decidable.

[1]Dipartimento di Matematica e Informatica, Università degli Studi di Catania, Città Universitaria, Viale A. Doria 6, 95152 Catania, Italy; e-mail: giugno@dmi.unict.it.

[2]Dipartimento di Informatica e Sistemistica, Università di Roma “La Sapienza”, Via Salaria 113, 00198 Rome, Italy; e-mail: lukasiewicz@dis.uniroma1.it. Alternate address: Institut für Informationssysteme, Technische Universität Wien, Favoritenstraße 9-11, 1040 Vienna, Austria; e-mail: lukasiewicz@kr.tuwien.ac.at.

Acknowledgements: This work was partially supported by a Marie Curie Individual Fellowship of the European Community (Disclaimer: The authors are solely responsible for information communicated and the European Commission is not responsible for any views or results expressed) and by the Austrian Science Fund Project N Z29-INF.

1 Introduction

The development of the *semantic web*, a vision for a future generation of the world wide web, aims at making web resources more easily accessible to automated processing by annotating web pages with machine-readable information on their content [2].

In the semantic web, ontologies are playing a central role as a source of shared and precisely defined terms that can be used in machine-readable semantic annotations of web pages. One important web ontology language is DAML+OIL [11, 12], which was developed by merging the DAML [9] and the OIL [5] web ontology language. A formal semantics and an automated reasoning support is provided to DAML+OIL through a mapping to the expressive description logic $\mathcal{SHOQ}(\mathbf{D})$ [15, 24] with the addition of inverse roles.

In the recent decades, dealing with probabilistic uncertainty has started to play an important role in database systems and knowledge representation and reasoning formalisms. We expect expressing and handling probabilistic knowledge to also play an important role in web ontology languages, which are essentially standardized languages for knowledge representation and reasoning, where significant research efforts are currently directed towards supporting query languages and large instances as in the field of databases [11, 12]. In particular, probabilistic web ontology languages may act as standardized tools to provide automated global web access to existing local data and knowledge base systems containing probabilistic information. It is thus not surprising that the OIL group writes in [13]:

- "a further level of extension" of OIL "could include modeling primitives such as defaults and fuzzy / probabilistic definitions".

However, to our knowledge, there are no such extensions of web ontology languages so far.

In this paper, we propose such an extension. We present a probabilistic extension of the description logic $\mathcal{SHOQ}(\mathbf{D})$, called P-$\mathcal{SHOQ}(\mathbf{D})$, which is based on the notion of probabilistic lexicographic entailment, a recent approach to probabilistic default reasoning [21, 22].

The description logic P-$\mathcal{SHOQ}(\mathbf{D})$ also allows to express default knowledge as a special case of generic probabilistic knowledge, where reasoning with such default knowledge is based on the sophisticated notion of lexicographic entailment by Lehmann [19].

There are several related approaches to probabilistic description logics in the literature [8, 17, 18], which can be classified according to the supported forms of probabilistic knowledge and the underlying probabilistic reasoning formalism. Heinsohn [8] presents a probabilistic extension of the description logic $\mathcal{ALC}$, which allows to represent generic probabilistic knowledge about concepts and roles, and which is essentially based on probabilistic reasoning in probabilistic logics, similar to [23, 1, 6, 20]. The work [8], however, does not allow for assertional (i.e., Abox) knowledge about concept and role instances. Also Jaeger [17] gives a probabilistic extension of the description logic $\mathcal{ALC}$, which allows for generic (resp., assertional) probabilistic knowledge about concepts and roles (resp., concept instances), but does not support probabilistic knowledge about role instances (but he mentions a possible extension in this direction). The uncertain reasoning formalism in [17] is essentially based on probabilistic reasoning in probabilistic logics, as the one in [8], but coupled with cross-entropy minimization to combine generic probabilistic knowledge with assertional probabilistic knowledge. The work by Koller et al. [18] gives a probabilistic generalization of the CLASSIC description logic. Like Heinsohn's work [8], it allows for generic probabilistic knowledge about concepts and roles, but does not support assertional knowledge about concept and role instances. However, differently from [8], it is based on inference in Bayesian networks as underlying probabilistic reasoning formalism. Note that approaches to fuzzy description logics [28, 27, 25, 26] are less closely related, as fuzzy uncertainty deals with vagueness, rather than ambiguity and imprecision.

The probabilistic description logic in this paper, differently from [8, 17, 18],

- is a probabilistic extension of the expressive description logic $\mathcal{SHOQ}(\mathbf{D})$, which provides a formal semantics and reasoning support for DAML+OIL (without inverse roles);
- allows to represent both generic probabilistic knowledge about concepts and roles, and also assertional probabilistic knowledge about concept and role instances;
- is based on probabilistic lexicographic entailment from recent approaches to probabilistic default reasoning [21, 22] as underlying probabilistic reasoning formalism.

The main contributions of this paper can be summarized as follows:

- We present a probabilistic extension of the description logic $\mathcal{SHOQ}(\mathbf{D})$, called P-$\mathcal{SHOQ}(\mathbf{D})$, which is based on the notion of probabilistic lexicographic entailment; this approach is motivated by recent advances to probabilistic default reasoning [21, 22]. Note especially that probabilistic lexicographic entailment has very nice properties for the use in probabilistic reasoning with generic and assertional probabilistic knowledge [21, 22].
- The description logic P-$\mathcal{SHOQ}(\mathbf{D})$ allows to express default knowledge as a special case of generic probabilistic knowledge, where reasoning with such default knowledge is based on the sophisticated notion of lexicographic entailment by Lehmann [19].
- We present sound and complete techniques for probabilistic reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$, which are based on reductions to classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$ and to linear programming, and which show in particular that reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$ is decidable. Note that due to the presence of individuals in P-$\mathcal{SHOQ}(\mathbf{D})$ (and also $\mathcal{SHOQ}(\mathbf{D})$), these techniques are technically more involved than the ones for probabilistic default reasoning in [21, 22].

The rest of this paper is organized as follows. In Section 2, we provide a motivating example. Section 3 recalls the description logic $\mathcal{SHOQ}(\mathbf{D})$. In Sections 4 and 5, we define our probabilistic extension P-$\mathcal{SHOQ}(\mathbf{D})$, and we give some illustrating examples, respectively. Section 6 describes sound and complete techniques for probabilistic reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$. In Section 7, we summarize the main results and give an outlook on future research. Note that detailed proofs of all results are given in the appendix.

2 Motivating Example

To illustrate the possible use of probabilistic ontologies in the semantic web, consider some medical knowledge about patients. It is advantageous to share such knowledge between hospitals or medical centers [10], for example, to follow up patients, to track medical history, and for case studies research. Furthermore, in medical knowledge, we often have to deal with probabilistic uncertainty.

For example, consider patient records related to cardiological illnesses. The patients may be classified according to their gender, as the probability that a man has a cardiological illness or a pacemaker is different from the corresponding probability associated with a woman. We may have the *default knowledge* that cardiological illnesses typically cause high blood pressure, but that pacemaker patients typically do not suffer from high blood pressure. We may have the *probabilistic knowledge* that the symptoms of a pacemaker patient are abnormal heart beat (arrhythmia) with a probability in $[.98, 1]$, chest pain with a probability in $[.9, 1]$, and breathing difficulties with a probability in $[.6, 1]$.

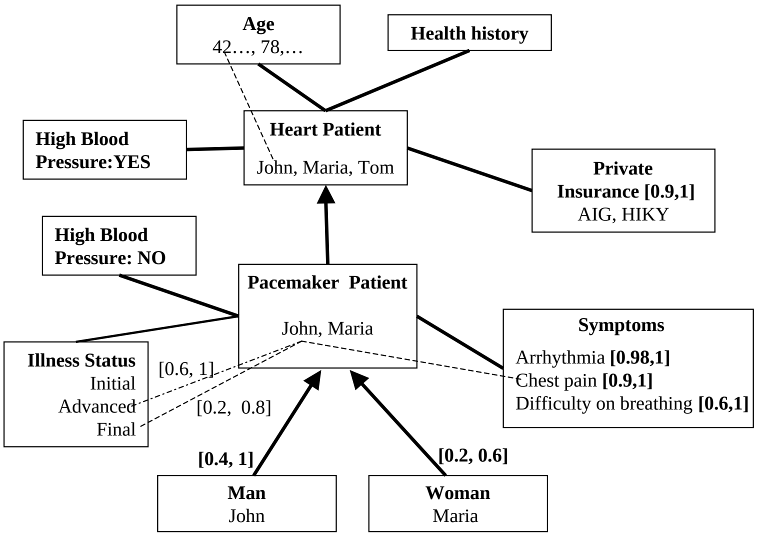

Figure 1: A probabilistic ontology.

In the description logic $\mathcal{SHOQ}(\mathbf{D})$, we distinguish between *concrete datatypes*, *individuals*, *concepts* (i.e., sets of individuals), and *roles* (i.e., binary relations on concepts, or between concepts and datatypes). For example, we may have concepts for patients, pacemaker patients, heart patients, female patients, male patients, symptoms, health insurances, and illness statuses. As individuals, we may have the heart patient John, the symptoms arrhythmia, chest pain, and breathing difficulties, and the illness statuses advanced and final. We may use roles to relate patients with their health insurances and with the status of their illnesses.

In P-$\mathcal{SHOQ}(\mathbf{D})$, we can express the following forms of probabilistic knowledge:

- The probabilistic knowledge that an instance of a concept is also an instance of another concept. E.g., the probability that a pacemaker patient is a man is in $[.4, 1]$.
- The probabilistic knowledge that an arbitrary instance of a concept is related to a given individual by a given role. E.g., the probability that a heart patient has a private insurance is in $[.9, 1]$.
- The probabilistic knowledge that an individual is an instance of a concept. E.g., the probability that difficulty breathing is a symptom of a pacemaker patient is in $[.6, 1]$, while the probability that chest pain is such a symptom is in $[.9, 1]$.
- The probabilistic knowledge that an individual is related to another individual by a role. E.g., the probability that the status of John's illness is final is in $[.2, .8]$.

3 $\mathcal{SHOQ}(\mathbf{D})$

In this section, we briefly recall the description logic $\mathcal{SHOQ}(\mathbf{D})$ from [15]. We first describe the syntax and semantics of $\mathcal{SHOQ}(\mathbf{D})$, and we then summarize some important reasoning tasks.

3.1 Syntax

We now recall the syntax of the description logic $\mathcal{SHOQ}(\mathbf{D})$. Informally, knowledge about concepts and roles is expressed in terminologies, which are finite sets of concept inclusion axioms, role inclusion axioms, and transitivity axioms. Concepts are constructed from primitive concepts and individuals using (i) the Boolean operators conjunction, disjunction, and negation, (ii) exists, value, atleast, and atmost restrictions on abstract roles, and (iii) datatype exists and datatype value restrictions on concrete roles.

We assume a set $\mathbf{D}$ of *concrete datatypes*. Every datatype $d \in \mathbf{D}$ is assigned a *domain* $\mathrm{dom}(d)$. We use $\mathrm{dom}(\mathbf{D})$ to denote the union of the domains $\mathrm{dom}(d)$ of all datatypes $d \in \mathbf{D}$. Let $\mathbf{C}$, $\mathbf{R}_A$, $\mathbf{R}_D$, and $\mathbf{I}$ be nonempty finite disjoint sets of *atomic concepts*, *abstract roles*, *concrete roles*, and *individuals*, respectively.

Concepts are inductively defined as follows. Every atomic concept from $\mathbf{C}$ is a concept. If o is an individual from $\mathbf{I}$, then $\{o\}$ is a concept. If C and D are concepts, then also $(C \sqcap D)$, $(C \sqcup D)$, and $\neg C$ (called *conjunction*, *disjunction*, and *negation*, respectively). If C is a concept, R is an abstract role from $\mathbf{R}_A$, and n is a nonnegative integer, then $\exists R.C$, $\forall R.C$, $\geq nR.C$, and $\leq nR.C$ are concepts (called *exists*, *value*, *atleast*, and *atmost restriction*, respectively). If T is a concrete role from $\mathbf{R}_D$, and d is a concrete datatype from $\mathbf{D}$, then $\exists T.d$ and $\forall T.d$ are concepts (called *datatype exists* and *datatype value*, respectively). We write $\top$ (resp., $\bot$) to abbreviate $C \sqcup \neg C$ (resp., $C \sqcap \neg C$), and we eliminate parentheses as usual.

A *concept inclusion axiom* is an expression of the form $C \sqsubseteq D$, where C and D are concepts. A *role inclusion axiom* is an expression $R \sqsubseteq S$, where either $R, S \in \mathbf{R}_A$ or $R, S \in \mathbf{R}_D$. A *transitivity axiom* has the form $\mathrm{Trans}(R)$, where $R \in \mathbf{R}_A$. A *terminological axiom* is either a concept inclusion axiom, a role inclusion axiom, or a transitivity axiom. A *terminology* $\mathcal{T}$ is a finite set of terminological axioms. A role R is called *simple* w.r.t. $\mathcal{T}$ iff for each role S, it holds that $S \sqsubseteq^{\star} R$ implies $\mathrm{Trans}(S) \notin \mathcal{T}$, where $\sqsubseteq^{\star}$ is the transitive and reflexive closure of $\sqsubseteq$ on $\mathcal{T}$.

It is important to point out that the above terminologies are expressive enough to also express knowledge about instances of concepts and of abstract roles: The knowledge that the individual o is an instance of the concept C can be expressed by the concept inclusion axiom $\{o\} \sqsubseteq C$; the knowledge that the pair of individuals (o_1, o_2) is an instance of the abstract role R can be expressed by the concept inclusion axiom $\{o_1\} \sqsubseteq \exists R.\{o_2\}$. We thus use $o \in C$ (resp., $(o_1, o_2) \in R$) to abbreviate $\{o\} \sqsubseteq C$ (resp., $\{o_1\} \sqsubseteq \exists R.\{o_2\}$). Note also that attributes (that is, functional roles) of concepts can be expressed as follows. To say that the concept D has the attribute A with the possible values $v_1, \ldots, v_k$, we can write $D \sqsubseteq {\geq}1A.C \sqcap {\leq}1A.C \sqcap \forall A.C$, where A is an abstract role from $\mathbf{R}_A$, and C is an atomic concept from $\mathbf{C}$ that is defined by $C \sqsubseteq \{v_1\} \sqcup \cdots \sqcup \{v_k\}$ and $\{v_1\} \sqcup \cdots \sqcup \{v_k\} \sqsubseteq C$, where $v_1, \ldots, v_k$ are individuals from $\mathbf{I}$.

3.2 Semantics

We now recall the semantics of $\mathcal{SHOQ}(\mathbf{D})$. Informally, interpretations with respect to abstract domains assign to each atomic concept a subset of the domain, to each individual an element of the domain, to each abstract role a binary relation on the abstract domain, and to each concrete role a binary relation between the abstract and the concrete domain. These interpretations are then extended to complex concepts and to concept inclusion axioms in the usual way. Moreover, the notions of satisfiability and of logical consequence for terminological axioms are also defined as usual.

An *interpretation* $\mathcal{I} = (\Delta, I)$ with respect to the set of concrete datatypes $\mathbf{D}$ consists of a nonempty *(abstract) domain* Δ and a mapping I that assigns to each atomic concept from $\mathbf{C}$ a subset of Δ, to each $\{o\}$ with $o \in \mathbf{I}$ a singleton subset of Δ, to each abstract role from $\mathbf{R}_A$ a subset of $\Delta \times \Delta$, and to each concrete role from $\mathbf{R}_D$ a subset of $\Delta \times \mathrm{dom}(\mathbf{D})$. The interpretation I is inductively extended to all concepts as follows (where $\#S$ denotes the cardinality of a set S):

- $I(C \sqcap D) = I(C) \cap I(D)$, $I(C \sqcup D) = I(C) \cup I(D)$, and $I(\neg C) = \Delta \setminus I(C)$,
- $I(\exists R.C) = \{x \in \Delta \mid \exists y\colon (x,y) \in I(R) \wedge y \in I(C)\}$,
- $I(\forall R.C) = \{x \in \Delta \mid \forall y\colon (x,y) \in I(R) \rightarrow y \in I(C)\}$,
- $I({\geq} nR.C) = \{x \in \Delta \mid \#(\{y \mid (x,y) \in I(R)\} \cap I(C)) \geq n\}$,
- $I({\leq} nR.C) = \{x \in \Delta \mid \#(\{y \mid (x,y) \in I(R)\} \cap I(C)) \leq n\}$,
- $I(\exists T.d) = \{x \in \Delta \mid \exists y\colon (x,y) \in I(T) \wedge y \in \mathrm{dom}(d)\}$,
- $I(\forall T.d) = \{x \in \Delta \mid \forall y\colon (x,y) \in I(T) \rightarrow y \in \mathrm{dom}(d)\}$.

The *satisfaction* of a terminological axiom F in $\mathcal{I}$, denoted $\mathcal{I} \models F$, is defined as follows:

- $\mathcal{I} \models C \sqsubseteq D$ iff $I(C) \subseteq I(D)$,
- $\mathcal{I} \models R \sqsubseteq S$ iff $I(R) \subseteq I(S)$,
- $\mathcal{I} \models \mathrm{Trans}(R)$ iff $I(R)$ is transitive.

The interpretation $\mathcal{I}$ *satisfies* a terminological axiom F, or $\mathcal{I}$ is a *model* of F, iff $\mathcal{I} \models F$. It *satisfies* a terminology $\mathcal{T}$, or $\mathcal{I}$ is a *model* of $\mathcal{T}$, denoted $\mathcal{I} \models \mathcal{T}$, iff $\mathcal{I} \models F$ for all $F \in \mathcal{T}$. The terminology $\mathcal{T}$ is *satisfiable* iff a model of $\mathcal{T}$ exists. A terminological axiom F is a *logical consequence* of $\mathcal{T}$, denoted $\mathcal{T} \models F$, iff every model of $\mathcal{T}$ is also a model of F.

3.3 Reasoning Tasks

We briefly summarize some important reasoning tasks in $\mathcal{SHOQ}(\mathbf{D})$. Informally, these tasks are to decide whether a terminology is satisfiable, whether a concept is satisfiable, whether a concept subsumes another concept, and whether an object (resp., pair of objects) is an instance of a concept (resp., an abstract role):

Terminology-Satisfiability: Given a terminology $\mathcal{T}$, decide whether $\mathcal{T}$ is satisfiable.

Concept-Satisfiability: Given a terminology $\mathcal{T}$ and a concept C, decide whether $\mathcal{T} \not\models C \sqsubseteq \bot$.

Concept-Subsumption: Given a terminology $\mathcal{T}$ and two concepts C and D, decide whether $\mathcal{T} \models C \sqsubseteq D$.

Concept-Membership: Given a terminology $\mathcal{T}$, $o \in \mathbf{I}$, and a concept C, decide whether $\mathcal{T} \models o \in C$.

Role-Membership: Given a terminology $\mathcal{T}$, $o_1, o_2 \in \mathbf{I}$, and $R \in \mathbf{R}_A$, decide whether $\mathcal{T} \models (o_1, o_2) \in R$.

It is not difficult to see that all the above problems can be reduced to Concept-Satisfiability and Concept-Subsumption, and that these two problems can be reduced to each other. In detail, Concept-Membership and Role-Membership are obviously special cases of Concept-Subsumption. Furthermore, Terminology-Satisfiability is a special case of Concept-Satisfiability, as a terminology $\mathcal{T}$ is satisfiable iff $\mathcal{T} \not\models \top \sqsubseteq \bot$. Finally, Concept-Satisfiability and Concept-Subsumption can be reduced to each other, as for every terminology $\mathcal{T}$ and for all concepts C and D, it holds that $\mathcal{T} \models C \sqcap \neg D \sqsubseteq \bot$ iff $\mathcal{T} \models C \sqsubseteq D$. These last two problems are decidable in $\mathcal{SHOQ}(\mathbf{D})$, if all atmost and atleast restrictions in $\mathcal{T}$ are restricted to simple abstract roles w.r.t. $\mathcal{T}$ [15].

4 P-$\mathcal{SHOQ}(\mathbf{D})$

In this section, we present the probabilistic description logic P-$\mathcal{SHOQ}(\mathbf{D})$, which is a probabilistic extension of $\mathcal{SHOQ}(\mathbf{D})$. We first define the syntax of P-$\mathcal{SHOQ}(\mathbf{D})$, where we have conditional constraints [20] to express probabilistic knowledge in addition to the terminological axioms of $\mathcal{SHOQ}(\mathbf{D})$. We then define the semantics of P-$\mathcal{SHOQ}(\mathbf{D})$, which is based on the notion of lexicographic entailment from probabilistic default reasoning; see especially [21, 22] for background, intuitions, and further examples. We finally summarize some important reasoning problems in P-$\mathcal{SHOQ}(\mathbf{D})$.

4.1 Syntax

We now define the notion of a probabilistic terminology. It is based on the language of conditional constraints [20], which encode interval restrictions for conditional probabilities over concepts. Every probabilistic terminology consists of a generic part, which expresses generic classical and probabilistic knowledge about concepts, and an assertional part, which represents classical and probabilistic knowledge about a set of individuals. In the sequel, we partition the set of individuals $\mathbf{I}$ into the set of *classical individuals* $\mathbf{I}_C$ and the set of *probabilistic individuals* $\mathbf{I}_P$. Intuitively, probabilistic individuals are those individuals in $\mathbf{I}$ for which we explicitly store some classical and probabilistic knowledge in a probabilistic terminology.

A *conditional constraint* is an expression of the form $(D|C)[l,u]$ with concepts C, D, and real numbers $l, u \in [0,1]$. A concept (resp., concept inclusion axiom, conditional constraint) is *generic* iff no probabilistic individual $o \in \mathbf{I}_P$ occurs in it. A concept inclusion axiom (resp., conditional constraint) is *assertional* for a probabilistic individual $o \in \mathbf{I}_P$ iff it is of the form $\{o\} \sqsubseteq D$ (resp., $(D|\{o\})[l,u]$), where D is generic. A *generic probabilistic terminology* (resp., an *assertional probabilistic terminology* for a probabilistic individual $o \in \mathbf{I}_P$) $\mathcal{P} = (\mathcal{T}, \mathcal{D})$ consists of a classical terminology $\mathcal{T}$ and a finite set of conditional constraints $\mathcal{D}$ such that every concept inclusion axiom in $\mathcal{T}$ and every conditional constraint in $\mathcal{D}$ is generic (resp., assertional for o). A *probabilistic terminology* $\mathcal{P} = (\mathcal{P}_g, (\mathcal{P}_o)_{o \in \mathbf{I}_P})$ with respect to $\mathbf{I}_P$ consists of a generic probabilistic terminology $\mathcal{P}_g$ and an assertional probabilistic terminology $\mathcal{P}_o$ for every $o \in \mathbf{I}_P$.

The different kinds of probabilistic knowledge that can be represented through conditional constraints in P-$\mathcal{SHOQ}(\mathbf{D})$ are briefly illustrated as follows:

- The probabilistic knowledge that "an instance of the concept C is also an instance of the concept D with a probability in $[l,u]$" can be expressed by $(D|C)[l,u]$.
- The probabilistic knowledge that "an arbitrary instance of the concept C is related to a given individual $o \in \mathbf{I}_C$ by a given role $R \in \mathbf{R}_A$ with a probability in $[l,u]$" can be expressed by $(\exists R.\{o\}|C)[l,u]$.
- The probabilistic knowledge that "the individual $o \in \mathbf{I}_P$ is an instance of the concept D with a probability in $[l,u]$" can be expressed by $(D|\{o\})[l,u]$.
- The probabilistic knowledge that "the individual $o \in \mathbf{I}_P$ is related to the individual $o' \in \mathbf{I}_C$ by the role $R \in \mathbf{R}_A$ with a probability in $[l,u]$" can be expressed by $(\exists R.\{o'\}|\{o\})[l,u]$.

4.2 Semantics

We now define the probabilistic semantics of P-$\mathcal{SHOQ}(\mathbf{D})$. We first generalize classical interpretations to probabilistic interpretations by adding a probability distribution over the abstract domain. We then define the satisfaction of terminological axioms and conditional constraints in probabilistic interpretations. We finally define the notions of consistency and entailment for probabilistic terminologies, which are based on the notions of consistency and lexicographic entailment in probabilistic default reasoning [21, 22].

A *probabilistic interpretation* $Pr = (\mathcal{I}, \mu)$ with respect to the set of concrete datatypes $\mathbf{D}$ consists of a classical interpretation $\mathcal{I} = (\Delta, I)$ with respect to $\mathbf{D}$ and a probability function μ on Δ (that is, a mapping $\mu\colon \Delta \rightarrow [0,1]$ such that all $\mu(o)$ with $o \in \Delta$ sum up to 1).

We now define the probability of a concept and the satisfaction of terminological axioms and conditional constraints in probabilistic interpretations as follows. The *probability* of a concept C in a probabilistic interpretation $Pr = (\mathcal{I}, \mu)$ with $\mathcal{I} = (\Delta, I)$, denoted $Pr(C)$, is the sum of all $\mu(o)$ such that $o \in I(C)$. For concepts C and D with $Pr(C) > 0$, we use the expression $Pr(D|C)$ to abbreviate $Pr(C \sqcap D)\ /\ Pr(C)$. We say Pr *satisfies* a conditional constraint $(D|C)[l,u]$, or Pr is a *model* of $(D|C)[l,u]$, denoted $Pr \models (D|C)[l,u]$, iff $Pr(D|C) \in [l,u]$. We say Pr *satisfies* a terminological axiom F, or Pr is a *model* of F, denoted $Pr \models F$, iff $\mathcal{I} \models F$. We say Pr *satisfies* a set of terminological axioms and conditional constraints $\mathcal{F}$, or Pr is a *model* of $\mathcal{F}$, denoted $Pr \models \mathcal{F}$, iff $Pr \models F$ for all $F \in \mathcal{F}$. We say $\mathcal{F}$ is *satisfiable* iff a model of $\mathcal{F}$ exists.

We next define the notion of consistency for probabilistic terminologies and generic probabilistic terminologies. We first give some preparative definitions from probabilistic default reasoning.

A probabilistic interpretation Pr *verifies* a conditional constraint $(D|C)[l,u]$ iff $Pr(C) = 1$ and $Pr \models (D|C)[l,u]$. We say Pr *falsifies* $(D|C)[l,u]$ iff $Pr(C) = 1$ and $Pr \not\models (D|C)[l,u]$. A set of conditional constraints $\mathcal{D}$ *tolerates* a conditional constraint F under a terminology $\mathcal{T}$ iff $\mathcal{T} \cup \mathcal{D}$ has a model that verifies F.

A generic probabilistic terminology $\mathcal{P}_g = (\mathcal{T}_g, \mathcal{D}_g)$ is *consistent* iff there exists an ordered partition $(\mathcal{D}_0, \ldots, \mathcal{D}_k)$ of $\mathcal{D}_g$ such that each $\mathcal{D}_i$ with $i \in \{0, \ldots, k\}$ is the set of all $F \in \mathcal{D}_g \setminus (\mathcal{D}_0 \cup \cdots \cup \mathcal{D}_{i-1})$ that are tolerated under $\mathcal{T}_g$ by $\mathcal{D}_g \setminus (\mathcal{D}_0 \cup \cdots \cup \mathcal{D}_{i-1})$. We call this ordered partition of $\mathcal{D}_g$ the *z-partition* of $\mathcal{P}_g$. A probabilistic terminology $\mathcal{P} = (\mathcal{P}_g, (\mathcal{P}_o)_{o \in \mathbf{I}_P})$, where $\mathcal{P}_g = (\mathcal{T}_g, \mathcal{D}_g)$ and $\mathcal{P}_o = (\mathcal{T}_o, \mathcal{D}_o)$ for all $o \in \mathbf{I}_P$, is *generically consistent* (or *g-consistent*) iff $\mathcal{P}_g$ is consistent. We say $\mathcal{P}$ is *consistent* iff $\mathcal{P}$ is g-consistent and $\mathcal{T}_g \cup \mathcal{T}_o \cup \mathcal{D}_o \cup \{(\{o\}|\top)[1,1]\}$ is satisfiable for all $o \in \mathbf{I}_P$.

We finally define the notion of lexicographic entailment of conditional constraints from probabilistic terminologies. In the rest of this subsection, let $\mathcal{P} = (\mathcal{P}_g, (\mathcal{P}_o)_{o \in \mathbf{I}_P})$, where $\mathcal{P}_g = (\mathcal{T}_g, \mathcal{D}_g)$ and $\mathcal{P}_o = (\mathcal{T}_o, \mathcal{D}_o)$ for all $o \in \mathbf{I}_P$, be a consistent probabilistic terminology.

We use the z-partition $(\mathcal{D}_0, \ldots, \mathcal{D}_k)$ of $\mathcal{P}_g$ to define a lexicographic preference relation on probabilistic interpretations as follows. For probabilistic interpretations Pr and Pr', we say Pr is *lexicographically preferable* (or *lex-preferable*) to Pr' iff some $i \in \{0, \ldots, k\}$ exists such that $\#(\{F \in \mathcal{D}_i \mid Pr \models F\}) > \#(\{F \in \mathcal{D}_i \mid Pr' \models F\})$ and $\#(\{F \in \mathcal{D}_j \mid Pr \models F\}) = \#(\{F \in \mathcal{D}_j \mid Pr' \models F\})$ for all $i < j \leq k$. A model Pr of a set of terminological axioms and conditional constraints $\mathcal{F}$ is a *lexicographically minimal model* (or *lex-minimal model*) of $\mathcal{F}$ iff no model of $\mathcal{F}$ is *lex*-preferable to Pr.

We now define the notion of lexicographic entailment for conditional constraints from sets of terminological axioms and conditional constraints under generic probabilistic terminologies as follows. A conditional constraint $(D|C)[l,u]$ is a *lexicographic consequence* (or *lex-consequence*) of a set of terminological axioms and conditional constraints $\mathcal{F}$ under $\mathcal{P}_g$, denoted $\mathcal{F} \mathrel{\|\!\!\sim} (D|C)[l,u]$ under $\mathcal{P}_g$, iff $Pr(D) \in [l,u]$ for every *lex*-minimal model Pr of $\mathcal{F} \cup \{(C|\top)[1,1]\}$. We say $(D|C)[l,u]$ is a *tight lexicographic consequence* (or *tight lex-consequence*) of $\mathcal{F}$ under $\mathcal{P}_g$, denoted $\mathcal{F} \mathrel{\|\!\!\sim}_{tight} (D|C)[l,u]$ under $\mathcal{P}_g$, iff l (resp., u) is the infimum (resp., supremum) of $Pr(D)$ subject to all *lex*-minimal models Pr of $\mathcal{F} \cup \{(C|\top)[1,1]\}$. Note that here we naturally define $l = 1$ and $u = 0$, when no such model Pr exists.

We are now ready to define which generic (resp., assertional) conditional constraints follow under lexicographic entailment from a probabilistic terminology. A generic conditional constraint F is a *lex-consequence* of $\mathcal{P}$, denoted $\mathcal{P} \mathrel{\|\!\!\sim} F$, iff $\emptyset \mathrel{\|\!\!\sim} F$ under $\mathcal{P}_g$. We say that F is a *tight lex-consequence* of $\mathcal{P}$, denoted $\mathcal{P} \mathrel{\|\!\!\sim}_{tight} F$, iff $\emptyset \mathrel{\|\!\!\sim}_{tight} F$ under $\mathcal{P}_g$. An assertional conditional constraint F for $o \in \mathbf{I}_P$ is a *lex-*

consequence of $\mathcal{P}$, denoted $\mathcal{P} \Vdash\!\sim F$, iff $\mathcal{T}_o \cup \mathcal{D}_o \Vdash\!\sim F$ under $\mathcal{P}_g$. We say F is a *tight lex-consequence* of $\mathcal{P}$, denoted $\mathcal{P} \Vdash\!\sim_{tight} F$, iff $\mathcal{T}_o \cup \mathcal{D}_o \Vdash\!\sim_{tight} F$ under $\mathcal{P}_g$.

4.3 Reasoning Tasks

We now summarize some important reasoning tasks in P-$\mathcal{SHOQ}(\mathbf{D})$. We first describe tasks related to the generic knowledge in probabilistic terminologies, namely (i) to decide whether a probabilistic terminology $\mathcal{P}$ is g-consistent, (ii) to decide whether a generic concept may have a positive probability under a g-consistent probabilistic terminology $\mathcal{P}$, (iii) to compute the tight interval under $\mathcal{P}$ for the probability that an individual is an instance of a generic concept D given that it is an instance of a generic concept C, and (iv) to compute the tight interval under $\mathcal{P}$ for the probability that an instance of a generic concept C is related to the individual $o \in \mathbf{I}_C$ by a role $R \in \mathbf{R}_A$:

P-Terminology-G-Consistency: Given a probabilistic terminology $\mathcal{P}$, decide whether $\mathcal{P}$ is g-consistent.

P-Concept-Satisfiability: Given a g-consistent probabilistic terminology $\mathcal{P}$ and a generic concept C, decide whether $\mathcal{P} \not\models (C|\top)[0,0]$.

P-Concept-Overlapping: Given a g-consistent probabilistic terminology $\mathcal{P}$ and generic concepts C, D, compute $l, u \in [0,1]$ such that $\mathcal{P} \Vdash\!\sim_{tight} (D|C)[l,u]$.

P-Concept-Role-Overlapping: Given a g-consistent $\mathcal{P}$, a generic concept C, an individual $o \in \mathbf{I}_C$, and an abstract role $R \in \mathbf{R}_A$, compute $l, u \in [0,1]$ such that $\mathcal{P} \Vdash\!\sim_{tight} (\exists R.\{o\}|C)[l,u]$.

We next give some reasoning tasks that are related to instances of concepts and abstract roles, and thus also concern the assertional knowledge in probabilistic terminologies. These reasoning tasks are (i) to decide whether a g-consistent probabilistic terminology is also consistent, (ii) to compute the tight interval under a consistent probabilistic terminology $\mathcal{P}$ for the probability that an individual $o \in \mathbf{I}_P$ is an instance of a generic concept D, and (iii) to compute the tight interval under $\mathcal{P}$ for the probability that an individual $o \in \mathbf{I}_P$ is related to an individual $o' \in \mathbf{I}_C$ by a role $R \in \mathbf{R}_A$:

P-Terminology-Consistency: Given a g-consistent probabilistic terminology $\mathcal{P}$, decide if $\mathcal{P}$ is consistent.

P-Concept-Membership: Given a consistent probabilistic terminology $\mathcal{P}$, an individual $o \in \mathbf{I}_P$, and a generic concept D, compute $l, u \in [0,1]$ such that $\mathcal{P} \Vdash\!\sim_{tight} (D|\{o\})[l,u]$.

P-Role-Membership: Given a consistent probabilistic terminology $\mathcal{P}$, individuals $o' \in \mathbf{I}_C$ and $o \in \mathbf{I}_P$, and an abstract role $R \in \mathbf{R}_A$, compute $l, u \in [0,1]$ such that $\mathcal{P} \Vdash\!\sim_{tight} (\exists R.\{o'\}|\{o\})[l,u]$.

In the sequel, we use **GCON** to denote P-Terminology-G-Consistency. Clearly, P-Terminology-Consistency is reducible to the problem of deciding whether a finite set of terminological axioms and conditional constraints is satisfiable, which we call **SAT**. It is then easy to see that P-Concept-Satisfiability is reducible to Concept-Satisfiability. Finally, P-Concept-Overlapping, P-Concept-Role-Overlapping, P-Concept-Membership, and P-Role-Membership can be reduced to computing tight *lex*-entailed intervals from finite sets of terminological axioms and conditional constraints under generic probabilistic terminologies, which we call **TLEXC**. Techniques for solving **SAT**, **GCON**, and **TLEXC** are described in Section 6.

5 Examples

The following example illustrates the inheritance of default knowledge along subconcept relationships.

Example 5.1 The strict knowledge "all pacemaker patients are heart patients" and the default knowledge "generally, heart patients have high blood pressure" can be represented by the following generic probabilistic terminology (where $\mathsf{hb\_pressure}$ is a binary attribute):

$$\mathcal{P}_g = (\{\mathsf{pm\_patient} \sqsubseteq \mathsf{h\_patient}\},\ \{(\exists \mathsf{hb\_pressure}.\{\mathsf{yes}\} \mid \mathsf{h\_patient})[1,1]\})\,.$$

It is then easy to see that $\emptyset \mathrel{\|\!\!\sim}_{tight} (\exists \mathsf{hb\_pressure}.\{\mathsf{yes}\} \mid \mathsf{h\_patient})[1,1]$ and $\emptyset \mathrel{\|\!\!\sim}_{tight} (\exists \mathsf{hb\_pressure}.\{\mathsf{yes}\} \mid \mathsf{pm\_patient})[1,1]$ under $\mathcal{P}_g$. That is, under lexicographic entailment we conclude "generally, heart patients have high blood pressure" and "generally, pacemaker patients have high blood pressure". That is, the property of having high blood pressure is inherited from the concept of all heart patients down to the subconcept of all pacemaker patients. □

The next example shows that default knowledge attached to more specific concepts overrides default knowledge inherited from less specific superconcepts.

Example 5.2 The strict knowledge "all pacemaker patients are heart patients" and the default knowledge "generally, heart patients have high blood pressure" and "generally, pacemaker patients do not have high blood pressure" can be expressed by

$$\begin{aligned}\mathcal{P}_g = {} & (\{\mathsf{pm\_patient} \sqsubseteq \mathsf{h\_patient}\},\\ & \{(\exists \mathsf{hb\_pressure}.\{\mathsf{yes}\} \mid \mathsf{h\_patient})[1,1],\ (\exists \mathsf{hb\_pressure}.\{\mathsf{no}\} \mid \mathsf{pm\_patient})[1,1]\})\,.\end{aligned}$$

It is then easy to see that $\emptyset \mathrel{\|\!\!\sim}_{tight} (\exists \mathsf{hb\_pressure}.\{\mathsf{yes}\} \mid \mathsf{h\_patient})[1,1]$ and $\emptyset \mathrel{\|\!\!\sim}_{tight} (\exists \mathsf{hb\_pressure}.\{\mathsf{no}\} \mid \mathsf{pm\_patient})[1,1]$ under $\mathcal{P}_g$. That is, under lexicographic entailment we conclude "generally, pacemaker patients do not have high blood pressure" and "generally, heart patients have high blood pressure". That is, even though the property of having high blood pressure is inherited from the concept of all heart patients down to the subconcept of all pacemaker patients, it is overridden by the property of not having high blood pressure of the more specific concept of all pacemaker patients. □

The following example illustrates probabilistic properties of concepts and the probabilistic membership of individuals to concepts.

Example 5.3 Consider the strict knowledge "all pacemaker patients are heart patients" and the generic probabilistic knowledge "a heart patient has a private insurance with a probability of at least 0.9". Moreover, consider the assertional probabilistic knowledge "John is a pacemaker patient with a probability of at least 0.8". This knowledge can be represented by the probabilistic terminology $\mathcal{P} = (\mathcal{P}_g, (\mathcal{P}_{\mathsf{John}}))$, where

$$\begin{aligned}\mathcal{P}_g &= (\{\mathsf{pm\_patient} \sqsubseteq \mathsf{h\_patient}\}, \{(\exists \mathsf{has\_p\_insurance}.\{\mathsf{yes}\} \mid \mathsf{h\_patient})[0.9,1]\})\,,\\ \mathcal{P}_{\mathsf{John}} &= (\emptyset, \{(\mathsf{pm\_patient} \mid \{\mathsf{John}\})[0.8,1]\})\,.\end{aligned}$$

Then, $\mathcal{P} \mathrel{\|\!\!\sim}_{tight} (\exists \mathsf{has\_p\_insurance}.\{\mathsf{yes}\} \mid \{\mathsf{John}\})[0.72,1]$. That is, under lexicographic entailment, we conclude "John has a private insurance with a probability of at least 0.72". □

6 Probabilistic Reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$

We now present techniques for solving the problems **SAT**, **GCON**, and **TLEXC**. These techniques are based on reductions to classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$ and to linear programming. They show in particular that probabilistic reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$ is decidable.

6.1 Preliminaries

Roughly speaking, the key idea behind developing algorithms for **SAT**, **GCON**, and **TLEXC** is to eliminate the classical interpretations $\mathcal{I}$ in probabilistic interpretations $Pr = (\mathcal{I}, \mu)$. This is done by using probability functions on sets $R \in \mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ of pairwise disjoint and exhaustive concepts instead of probability functions on the abstract domain.

We start by defining a set $R_{\mathcal{T}}(\mathcal{F})$ of pairwise disjoint and exhaustive concepts for a classical terminology $\mathcal{T}$ and a set of conditional constraints $\mathcal{F} = \{F_1, \ldots, F_n\}$ as follows. Let $R_{\mathcal{T}}(\mathcal{F})$ be the set of all mappings r that assign to each $F_i = (D_i|C_i)[l_i, u_i] \in \mathcal{F}$ a member of $\{D_i \sqcap C_i, \neg D_i \sqcap C_i, \neg C_i\}$, such that $\mathcal{T} \not\models r(F_1) \sqcap \cdots \sqcap r(F_n) \sqsubseteq \bot$. For such mappings r, we use $\sqcap r$ to abbreviate $r(F_1) \sqcap \cdots \sqcap r(F_n)$. For such mappings r and concepts C, we use $r \models C$ to abbreviate $\emptyset \models \sqcap r \sqsubseteq C$.

We next define a set $\mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ of subsets of $R_{\mathcal{T}}(\mathcal{F})$ such that the models Pr of $\mathcal{T}$ correspond to the probability functions on all $R \in \mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$: We denote by $\mathcal{R}_{\mathcal{T}}(\mathcal{F})$ the set of all $R \subseteq R_{\mathcal{T}}(\mathcal{F})$ such that $\mathcal{T} \cup \{\{o_r\} \sqsubseteq \sqcap r \mid r \in R\}$ is satisfiable, where o_r is a new individual in $\mathbf{I}$ for every $r \in R_{\mathcal{T}}(\mathcal{F})$; we then denote by $\mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ the set of all maximal elements in $\mathcal{R}_{\mathcal{T}}(\mathcal{F})$ with respect to set inclusion.

Observe that $\mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ can be computed by classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$. It is also important to point out that, due to the concepts $\{o\}$ with individuals $o \in \mathbf{I}$ (also called *nominals*) in P-$\mathcal{SHOQ}(\mathbf{D})$ (and also $\mathcal{SHOQ}(\mathbf{D})$), it is in general not sufficient to define probability functions only on the set $R_{\mathcal{T}}(\mathcal{F})$.

6.2 Satisfiability

The following theorem shows that **SAT** can be reduced to deciding whether a system of linear constraints over a set of variables that corresponds to some $R \in \mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ is solvable. As $\mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ can be computed by classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$, this shows that **SAT** can be reduced to classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$ and to deciding whether a system of linear constraints is solvable. The theorem can be proved by showing that the models Pr of $\mathcal{T}$ correspond to the probability functions on all $R \in \mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$.

Theorem 6.1 *Let $\mathcal{T}$ be a classical terminology, and let $\mathcal{F}$ be a finite set of conditional constraints. Then, $\mathcal{T} \cup \mathcal{F}$ is satisfiable iff the system of linear constraints* (1) *in Fig. 2 over the variables y_r $(r \in R)$ is solvable for some $R \in \mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$.*

6.3 G-Consistency

The problem **GCON** can be reduced to **SAT**, as Algorithm z-partition in Fig. 3 shows, which decides whether a generic probabilistic terminology $\mathcal{P}_g$ is consistent. If this is the case, then z-partition returns the z-partition of $\mathcal{P}_g$, otherwise *nil*. In Step 5 of z-partition, a number of instances of **SAT** must be solved. Note that Algorithm z-partition is essentially a reformulation of an algorithm for deciding ε-consistency in default reasoning from conditional knowledge bases by Goldszmidt and Pearl [7].

$$\begin{aligned}
\sum_{r\in R,\, r\models\neg D\sqcap C} -l\, y_r + \sum_{r\in R,\, r\models D\sqcap C} (1-l)\, y_r &\geq 0 \quad \text{(for all } (D|C)[l,u]\in\mathcal{F},\ l>0)\\
\sum_{r\in R,\, r\models\neg D\sqcap C} u\, y_r + \sum_{r\in R,\, r\models D\sqcap C} (u-1)\, y_r &\geq 0 \quad \text{(for all } (D|C)[l,u]\in\mathcal{F},\ u<1)\\
\sum_{r\in R} y_r &= 1\\
y_r &\geq 0 \quad \text{(for all } r\in R)
\end{aligned} \tag{1}$$

Figure 2: System of linear constraints for Theorems 6.1 and 6.2.

6.4 Tight Logical Consequence

We will reduce the problem **TLEXC** to the problems **SAT** and **TLC**, where **TLC** is the problem of, given a concept B and a finite set of terminological axioms and conditional constraints $\mathcal{F}$, computing the real numbers $l, u \in [0,1]$ such that $(B|\top)[l,u]$ is a tight logical consequence of $\mathcal{F}$. We now formally define the notion of logical entailment for probabilistic terminologies, and then show how **TLC** can be reduced to classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$ and to linear programming.

A conditional constraint $(D|C)[l,u]$ is a *logical consequence* of a set of terminological axioms and conditional constraints $\mathcal{F}$, denoted $\mathcal{F} \models (D|C)[l,u]$, iff each model of $\mathcal{F}$ is also a model of $(D|C)[l,u]$. It is a *tight logical consequence* of $\mathcal{F}$, denoted $\mathcal{F} \models_{tight} (D|C)[l,u]$, iff l (resp., u) is the infimum (resp., supremum) of $Pr(D|C)$ subject to all models Pr of $\mathcal{F}$ with $Pr(C) > 0$.

The following theorem shows that the problem **TLC** is reducible to classical reasoning in $\mathcal{SHOQ}(\mathbf{D})$ and to computing the optimal values of linear programs.

Theorem 6.2 *Let $\mathcal{T}$ be a classical terminology, let $\mathcal{F}$ be a finite set of conditional constraints, and let B be a concept. Assume $\mathcal{T} \cup \mathcal{F}$ is satisfiable. Then, l (resp., u) such that $\mathcal{T} \cup \mathcal{F} \models_{tight} (B|\top)[l,u]$ is given by the minimum of l' (resp., maximum of u') subject to l' (resp., u') being the optimal value of the following linear program over the variables y_r $(r\in R)$ and $R \in \mathrm{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F} \cup \{(B|\top)[0,1]\}))$:*

$$\textit{minimize (resp., maximize)} \sum_{r\in R,\, r\models B} y_r \quad \textit{subject to } (1)\,. \tag{2}$$

6.5 Tight Lexicographic Consequence

We now show how **TLEXC** can be reduced to the problems **SAT** and **TLC**.

In the sequel, let $\mathcal{P}_g = (\mathcal{T}_g, \mathcal{D}_g)$ be a consistent generic probabilistic terminology, and let $(\mathcal{D}_0, \ldots, \mathcal{D}_k)$ be its z-partition (which can be computed using Algorithm z-partition in Fig. 3). Let $\mathcal{F}$ be a finite set of terminological axioms and conditional constraints. The key idea behind the reduction is that a set $\overline{\mathcal{D}}$ of subsets of $\mathcal{D}_g$ exists such that $\mathcal{F} \mathrel{|\!\!\sim} (D|C)[l,u]$ under $\mathcal{P}_g$ iff $\mathcal{T}_g \cup \mathcal{D}' \cup \mathcal{F} \cup \{(C|\top)[1,1]\} \models (D|\top)[l,u]$ for all $\mathcal{D}' \in \overline{\mathcal{D}}$. We need some preparative definitions as follows.

We say $\mathcal{D}' \subseteq \mathcal{D}_g$ is *lex-preferable* to $\mathcal{D}'' \subseteq \mathcal{D}_g$ iff some $i \in \{0, \ldots, k\}$ exists such that $|\mathcal{D}' \cap \mathcal{D}_i| > |\mathcal{D}'' \cap \mathcal{D}_i|$ and $|\mathcal{D}' \cap \mathcal{D}_j| = |\mathcal{D}'' \cap \mathcal{D}_j|$ for all $i < j \leq k$. We say $\mathcal{D}'$ is *lex-minimal* in $\mathcal{S} \subseteq \{S \mid S \subseteq \mathcal{D}_g\}$ iff $\mathcal{D}' \in \mathcal{S}$ and no $\mathcal{D}'' \in \mathcal{S}$ is *lex*-preferable to $\mathcal{D}'$.

Algorithm z-partition

Input: Generic probabilistic terminology $\mathcal{P}_g = (\mathcal{T}_g, \mathcal{D}_g)$ with $\mathcal{D}_g \neq \emptyset$.
Output: z-partition of $\mathcal{P}_g$, if $\mathcal{P}_g$ is consistent, otherwise *nil*.

$\mathcal{H} := \mathcal{D}_g$;
$i := -1$;
repeat
  $i := i + 1$
  $\mathcal{D}[i] := \{C \in \mathcal{H} \mid C$ is tolerated under $\mathcal{T}_g$ by $\mathcal{H}\}$;
  $\mathcal{H} := \mathcal{H} \setminus \mathcal{D}[i]$
until $\mathcal{H} = \emptyset$ **or** $\mathcal{D}[i] = \emptyset$;
if $\mathcal{H} = \emptyset$ **then return** $(\mathcal{D}[0], \ldots, \mathcal{D}[i])$
  else return *nil*.

Figure 3: Algorithm z-partition.

Algorithm tight-lex-consequence

Input: Consistent generic probabilistic terminology $\mathcal{P}_g = (\mathcal{T}_g, \mathcal{D}_g)$, set of terminological axioms and conditional constraints $\mathcal{F}$, and two concepts C and D.
Output: Pair of real numbers $(l, u) \in [0, 1]^2$ such that $\mathcal{F} \Vert\!\sim_{tight} (D|C)[l, u]$ under $\mathcal{P}_g$.
Notation: $(\mathcal{D}_0, \ldots, \mathcal{D}_k)$ denotes the z-partition of $\mathcal{P}_g$.

$\mathcal{R} := \mathcal{T}_g \cup \mathcal{F} \cup \{(C|\top)[1, 1]\}$;
if $\mathcal{R}$ is unsatisfiable **then return** $(1, 0)$;
$\overline{\mathcal{H}} := \{\emptyset\}$;
for $j := k$ **downto** 0 **do begin**
  $n := 0$;
  $\overline{\mathcal{H}}' := \emptyset$;
  for each $\mathcal{G} \subseteq \mathcal{D}_j$ and $\mathcal{H} \in \overline{\mathcal{H}}$ **do**
    if $\mathcal{R} \cup \mathcal{G} \cup \mathcal{H}$ is satisfiable **then**
      if $n = |\mathcal{G}|$ **then** $\overline{\mathcal{H}}' := \overline{\mathcal{H}}' \cup \{\mathcal{G} \cup \mathcal{H}\}$
        else if $n < |\mathcal{G}|$ **then begin**
          $\overline{\mathcal{H}}' := \{\mathcal{G} \cup \mathcal{H}\}$;
          $n := |\mathcal{G}|$
        end;
  $\overline{\mathcal{H}} := \overline{\mathcal{H}}'$;
end;
$(l, u) := (1, 0)$;
for each $\mathcal{H} \in \overline{\mathcal{H}}$ **do begin**
  compute $c, d \in [0, 1]$ s.t. $\mathcal{R} \cup \mathcal{H} \models_{tight} (D|\top)[c, d]$;
  $(l, u) := (\min(l, c), \max(u, d))$
end;
return (l, u).

Figure 4: Algorithm tight-lex-consequence.

The following theorem shows how tight *lex*-consequences can be characterized through tight logical consequences. It follows from a similar result in [22].

Theorem 6.3 *Let $\mathcal{P}_g = (\mathcal{T}_g, \mathcal{D}_g)$ be a consistent generic probabilistic terminology, let $\mathcal{F}$ be a finite set of terminological axioms and conditional constraints, and let C, D be two concepts. Let $\overline{\mathcal{D}}$ be the set of all lex-minimal elements in the set of all $\mathcal{D}' \subseteq \mathcal{D}_g$ such that $\mathcal{T}_g \cup \mathcal{D}' \cup \mathcal{F} \cup \{(C|\top)[1,1]\}$ is satisfiable. Then,*

(a) If $\overline{\mathcal{D}} = \emptyset$, then $\mathcal{F} \Vdash\!\sim_{tight} (D|C)[1,0]$ under $\mathcal{P}_g$.

(b) If $\overline{\mathcal{D}} \neq \emptyset$, then $\mathcal{F} \Vdash\!\sim_{tight} (D|C)[l,u]$ under $\mathcal{P}_g$, where $l = \min l'$ (resp., $u = \max u'$) subject to $\mathcal{T}_g \cup \mathcal{D}' \cup \mathcal{F} \cup \{(C|\top)[1,1]\} \models_{tight} (D|\top)[l',u']$ and $\mathcal{D}' \in \overline{\mathcal{D}}$.

Based on this result, Algorithm tight-lex-consequence in Fig. 4 computes tight lexicographically entailed intervals, by first (i) computing $\overline{\mathcal{D}}$, which is done by solving a number of instances of **SAT** in steps 3–15, and then (ii) computing tight logically entailed intervals, which is done by solving instances of **TLC** in steps 16–20.

7 Summary and Outlook

The main motivation behind this work was to develop a probabilistic extension of DAML+OIL for representing and reasoning with probabilistic ontologies in the semantic web. To this end, we worked out a probabilistic extension of $\mathcal{SHOQ}(\mathbf{D})$, which is the description logic that provides a formal semantics and a reasoning support for DAML+OIL (without inverse roles). The resulting new probabilistic description logic P-$\mathcal{SHOQ}(\mathbf{D})$ is based on the notion of probabilistic lexicographic entailment from probabilistic default reasoning. It allows to express rich probabilistic knowledge about concepts and instances, as well as default knowledge about concepts. We also presented sound and complete reasoning techniques for P-$\mathcal{SHOQ}(\mathbf{D})$, which show in particular that reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$ is decidable.

An interesting topic of future research is to analyze the computational complexity of probabilistic reasoning in P-$\mathcal{SHOQ}(\mathbf{D})$. Another issue for further work is to define a probabilistic extension of $\mathcal{SHOQ}(\mathbf{D_n})$ [24], which is a recent generalization of $\mathcal{SHOQ}(\mathbf{D})$ that also supports n-ary datatype predicates as well as datatype number restrictions. In particular, such an extension may also allow for expressing probabilistic knowledge about how instances of concepts are related to datatype values by roles. Finally, it would also be very interesting to allow for complex types and to develop more complex probabilistic query languages (e.g., similar to [3, 4]).

A Appendix: Proofs for Section 6

Consider a classical terminology $\mathcal{T}$, and a finite set of conditional constraints $\mathcal{F}$. The following result shows that the models $Pr = (\mathcal{I}, \mu)$ of $\mathcal{T}$ correspond to the probability functions on all $R \in \mathrm{Max}(\mathcal{R}_\mathcal{T}(\mathcal{F}))$.

Theorem A.1 *Let $\mathcal{T}$ be a classical terminology, and let $\mathcal{F}$ be a finite set of conditional constraints. Then,*

(a) For every model $Pr = ((\Delta, I), \mu)$ of $\mathcal{T}$, a probability function $\overline{\mu}$ on some $R \in \mathrm{Max}(\mathcal{R}_\mathcal{T}(\mathcal{F}))$ exists such that $Pr(\sqcap r) = \overline{\mu}(r)$ for all $r \in R$.

(b) *For every probability function* $\overline{\mu}$ *on some* $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$*, a model* $Pr = ((\Delta, I), \mu)$ *of* $\mathcal{T}$ *exists such that* $Pr(\sqcap r) = \overline{\mu}(r)$ *for all* $r \in R$.

Proof. (a) Let $Pr = ((\Delta, I), \mu)$ be a model of $\mathcal{T}$. We then define $R' = \{r \in R_{\mathcal{T}}(\mathcal{F}) \mid I(\sqcap r) \neq \emptyset\}$. Observe that $R' \in \mathcal{R}_{\mathcal{T}}(\mathcal{F})$, as Pr is a model of $\mathcal{T} \cup \{\{o_r\} \sqsubseteq \sqcap r \mid r \in R'\}$. Hence, some $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ exists such that $R' \subseteq R$. We then define the probability function $\overline{\mu}$ on R by $\overline{\mu}(r) = Pr(\sqcap r)$ for all $r \in R'$ and by $\overline{\mu}(r) = 0$ for all $r \in R \setminus R'$. It is now easy to see that $Pr(\sqcap r) = \overline{\mu}(r)$ for all $r \in R$.

(b) Let $\overline{\mu}$ be a probability function on some $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$. Hence, $\mathcal{T} \cup \{\{o_r\} \sqsubseteq \sqcap r \mid r \in R\}$ is satisfiable. That is, a model $\mathcal{I} = (\Delta, I)$ of $\mathcal{T}$ exists such that $I(\sqcap r) \neq \emptyset$ for all $r \in R$. We now define the probability function μ on Δ by $\mu(\delta) = \overline{\mu}(r) \,/\, |I(\sqcap r)|$ for all $\delta \in \Delta$ such that $\delta \in I(\sqcap r)$ for some $r \in R$ and by $\mu(\delta) = 0$ for all $\delta \in \Delta$ such that $\delta \notin I(\sqcap r)$ for all $r \in R$. Then, $Pr(\sqcap r) = \overline{\mu}(r)$ for all $r \in R$. □

The following example shows that, due to the concepts $\{o\}$ with individuals $o \in \mathbf{I}$, it is in general not the case that the models $Pr = (\mathcal{I}, \mu)$ of $\mathcal{T}$ correspond to the probability functions on $R_{\mathcal{T}}(\mathcal{F})$.

Example A.2 Consider the following classical terminology $\mathcal{T}$ over $\mathbf{C} = \{C, C_1, C_2\}$ and $\mathbf{I} = \{o\}$:

$$\mathcal{T} \;=\; \{C_1 \sqcap C_2 \sqsubseteq \bot,\ C \sqsubseteq C_1 \sqcup C_2,\ C_1 \sqcup C_2 \sqsubseteq C,\ C \sqsubseteq \{o\},\ \{o\} \sqsubseteq C\}\,.$$

Observe now that for every model $\mathcal{I} = (\Delta, I)$ of $\mathcal{T}$, it holds that $I(C_1) \cap I(C_2) = \emptyset$ and that $I(C) = I(C_1) \cup I(C_2) = I(\{o\}) = \{\delta\}$ for some $\delta \in \Delta$. Hence, it holds either $I(C_1) = \emptyset$ and $I(C_2) = \{\delta\}$ for some $\delta \in \Delta$, or $I(C_2) = \emptyset$ and $I(C_1) = \{\delta\}$ for some $\delta \in \Delta$.

Consider now the set of conditional constraints $\mathcal{F} = \{(C_1|C)[l, u]\}$. It then holds $R_{\mathcal{T}}(\mathcal{F}) = \{\neg C,\ C \sqcap C_1,\ C \sqcap \neg C_1\}$, $\mathcal{R}_{\mathcal{T}}(\mathcal{F}) = R_{\mathcal{T}}(\mathcal{F}) \cup \{\{\neg C,\ C \sqcap C_1\},\ \{\neg C,\ C \sqcap \neg C_1\}\}$, and $\text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F})) = \{\{\neg C,\ C \sqcap C_1\},\ \{\neg C,\ C \sqcap \neg C_1\}\}$. By Theorem A.1, every model $Pr = (\Delta, \mu)$ of $\mathcal{T}$ corresponds to a probability function on some $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$. But it does not hold that every model $Pr = (\Delta, \mu)$ of $\mathcal{T}$ corresponds to a probability function on $R_{\mathcal{T}}(\mathcal{F})$. For example, the probability function μ that is defined by $\mu(r) = 1/3$ for all $r \in R_{\mathcal{T}}(\mathcal{F})$ does not have any corresponding model Pr of $\mathcal{T}$. □

Proof of Theorem 6.1. Recall that $\mathcal{T} \cup \mathcal{F}$ is satisfiable iff there exists a model Pr of $\mathcal{T}$ that also satisfies $\mathcal{F}$. By Theorem A.1, this is equivalent to the existence of a probability function $\overline{\mu}$ on some $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$ that satisfies $\mathcal{F}$. It is well-known that the latter is equivalent to the system of linear constraints (1) in Fig. 2 over the variables y_r $(r \in R)$ being solvable for some $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F}))$. □

Proof of Theorem 6.2. Recall that $\mathcal{T} \cup \mathcal{F} \models_{tight} (B|\top)[l, u]$ iff l (resp., u) is the infimum (resp., supremum) of $Pr(B)$ subject to all models Pr of $\mathcal{T} \cup \mathcal{F}$. By Theorem A.1, this is equivalent to l (resp., u) being the infimum (resp., supremum) of $\sum_{r \in R,\, r \models B} \overline{\mu}(r)$ subject to all probability functions $\overline{\mu}$ on any $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F} \cup \{(B|\top)[0, 1]\}))$ satisfying $\mathcal{F}$. It is now well-known that we can equivalently say that l (resp., u) is the minimum of l' (resp., maximum of u') subject to l' (resp., u') being the optimal value of the linear program in (2) over the variables y_r $(r \in R)$ and $R \in \text{Max}(\mathcal{R}_{\mathcal{T}}(\mathcal{F} \cup \{(B|\top)[0, 1]\}))$. □

References

[1] S. Amarger, D. Dubois, and H. Prade. Constraint propagation with imprecise conditional probabilities. In *Proceedings UAI-91*, pages 26–34, 1991.

[2] T. Berners-Lee. *Weaving the Web*. Harper, San Francisco, 1999.

[3] T. Eiter, J. J. Lu, T. Lukasiewicz, and V. S. Subrahmanian. Probabilistic object bases. *ACM Transactions on Database Systems*, 26(3):264–312, 2001.

[4] T. Eiter, T. Lukasiewicz, and M. Walter. A data model and algebra for probabilistic complex values. *Ann. Math. Artif. Intell.*, 33(2-4):205–252, 2001.

[5] D. Fensel, F. van Harmelen, I. Horrocks, D. L. McGuiness, and P. F. Patel-Schneider. OIL: An ontology infrastructure for the semantic web. *IEEE Intelligent Systems*, 16(2):38–45, 2001.

[6] A. M. Frisch and P. Haddawy. Anytime deduction for probabilistic logic. *Artif. Intell.*, 69:93–122, 1994.

[7] M. Goldszmidt and J. Pearl. On the consistency of defeasible databases. *Artif. Intell.*, 52(2):121–149, 1991.

[8] J. Heinsohn. Probabilistic description logics. In *Proceedings UAI-94*, pages 311–318, 1994.

[9] J. Hendler and D. L. McGuiness. The DARPA agent markup language. *IEEE Intelligent Systems*, 15(6):67–73, 2000.

[10] Health level seven, 1997. www.hl7.org.

[11] I. Horrocks. DAML+OIL: A description logic for the semantic web. *IEEE Bulletin of the Technical Committee on Data Engineering*, 25(1):4–9, 2002.

[12] I. Horrocks. DAML+OIL: A reason-able web ontology language. In *Proceedings EDBT-02*, volume 2287 of *LNCS*, pages 2–13. Springer, 2002.

[13] I. Horrocks, D. Fensel, J. Broekstra, S. Decker, M. Erdmann, C. Goble, F. van Harmelen, M. Klein, S. Staab, R. Studer, and E. Motta. OIL: The ontology inference layer. Technical Report IR-479, Vrije Universiteit Amsterdam, 2000.

[14] I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen. Reviewing the design of DAML+OIL: An ontology language for the semantic web. In *Proceedings AAAI-02*, 2002. To appear.

[15] I. Horrocks and U. Sattler. Ontology reasoning in the $\mathcal{SHOQ}(\mathbf{D})$ description logic. In *Proceedings IJCAI-01*, pages 199–204, 2001.

[16] I. Horrocks, U. Sattler, and S. Tobies. Practical reasoning for expressive description logics. In *Proceedings LPAR-99*, volume 1705 of *LNCS*, pages 161–180. Springer, 1999.

[17] M. Jaeger. Probabilistic reasoning in terminological logics. In *Proceedings KR-94*, pages 305–316, 1994.

[18] D. Koller, A. Levy, and A. Pfeffer. P-CLASSIC: A tractable probabilistic description logic. In *Proceedings AAAI-97*, pages 390–397, 1997.

[19] D. Lehmann. Another perspective on default reasoning. *Ann. Math. Artif. Intell.*, 15(1):61–82, 1995.

[20] T. Lukasiewicz. Probabilistic deduction with conditional constraints over basic events. *J. Artif. Intell. Res.*, 10:199–241, 1999.

[21] T. Lukasiewicz. Probabilistic logic programming under inheritance with overriding. In *Proceedings UAI-01*, pages 329–336, 2001.

[22] T. Lukasiewicz. Probabilistic default reasoning with conditional constraints. *Ann. Math. Artif. Intell.*, 34(1–3):35–88, 2002.

[23] N. J. Nilsson. Probabilistic logic. *Artif. Intell.*, 28:71–88, 1986.

[24] J. Z. Pan and I. Horrocks. Semantic web ontology reasoning in the $\mathcal{SHOQ}(\mathbf{D_n})$ description logic. In *Proceedings DL-02*, 2002.

[25] U. Straccia. A fuzzy description logic. In *Proceedings AAAI-98*, pages 594–599, 1998.

[26] U. Straccia. Reasoning within fuzzy description logics. *J. Artif. Intell. Res.*, 14:137–166, 2001.

[27] C. B. Tresp and R. Molitor. A description logic for vague knowledge. In *Proceedings ECAI-98*, pages 361–365, 1998.

[28] J. Yen. Generalizing term subsumption languages to fuzzy logic. In *Proceedings IJCAI-91*, pages 472–477, 1991.