Query Answering in Bayesian Description Logics

İsmail İlkan Ceylan*

Theoretical Computer Science, TU Dresden, Germany
ceylan@tcs.inf.tu-dresden.de

Abstract. The Bayesian Description Logic (BDL) $\mathcal{BEL}$ is a probabilistic DL, which extends the lightweight DL $\mathcal{EL}$ by defining a joint probability distribution over $\mathcal{EL}$ axioms with the help of a Bayesian network (BN). In the recent work, extensions of standard logical reasoning tasks in $\mathcal{BEL}$ are shown to be reducible to inferences in BNs.
This work concentrates on a more general reasoning task, namely on conjunctive query answering in $\mathcal{BEL}$ where every query is associated to a probability leading to different reasoning problems. In particular, we study the probabilistic query entailment, top-k answers, and top-k contexts as reasoning problems. Our complexity analysis suggests that all of these problems are tractable under certain assumptions.

1 Introduction

Description Logics (DLs) [3], as a successful family of knowledge representation (KR) formalisms, have been employed in various application domains such as conceptual modeling, databases, bio-medical ontologies, natural language processing, configuration, and the semantic web[1]. Arguably, all these domains, as is real world, are subject to imprecision; may it be an assertion about an individual or a terminological statement, it often comes along with a degree of uncertainty.

The fact that classical DLs had severe limitations in representing and reasoning under uncertainty led to a body of work [20] tailored towards this goal. Several extensions to DLs have been proposed with different characteristics in terms of their logical expressivity, their semantics, and their independence assumptions.

BDLs [6] have been proposed as a means of representing the uncertainty over DL axioms that are being asserted. In BDLs, every axiom is associated with a probability, which is encoded with the help of a BN. This family of logics provides a compact and easy way of encoding probabilities over DL axioms. Two important features of BDLs are that they do not force any independence assumptions, and they are based on the so-called multiple world semantics.

The focus of this work is the DL $\mathcal{BEL}$ [7], a Bayesian extension of the lightweight DL $\mathcal{EL}$ [2] for which several probabilistic reasoning tasks have been

* Supported by DFG within the Research Training Group "RoSI" (GRK 1907).
[1] http://www.w3.org/TR/owl2-overview/

Table 1: Syntax and Semantics of $\mathcal{EL}$

Name	Syntax	Semantics
Top	$\top$	$\Delta^{\mathcal{I}}$
Conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Exist. Rest.	$\exists r.C$	$\{d \mid \exists e \in \Delta^{\mathcal{I}} : (d,e) \in r^{\mathcal{I}}, e \in C^{\mathcal{I}}\}$
GCI	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
Concept assertion	$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
Role assertion	$r(a,b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$

studied such as the probabilistic entailment, or finding most likely context (sub-ontology) for an entailment. In fact, tight complexity bounds have been obtained for these problems [8].

Nevertheless, problems related to query answering, and in particular conjunctive query (CQ) answering, has not been studied in the context of BDLs, so far. In this paper, we close this gap and focus on i) probabilistic query entailment: "What is the probability of a query to be entailed?" ii) probabilistic query answering: "What are the *top-k answers* to a query?" and finally iii) the most likely context: "What are the *top-k contexts* that entail a query?"

Consequently, we argue that these problems generalize the reasoning problems that have been considered so far. Unsurprisingly, reasoning in $\mathcal{BEL}$ is intractible as is CQ answering in $\mathcal{EL}$ and inference in BNs. Further analysis shows that tractability can be regained by fixing the BN and the query.

2 Conjunctive Query Answering in $\mathcal{EL}$

We briefly review the DL $\mathcal{EL}$ [5] and query answering in $\mathcal{EL}$, which constitute the basis of this paper. Formally, let $\mathsf{N_I}$, $\mathsf{N_C}$ and $\mathsf{N_R}$ be disjoint sets of *individual-*, *concept-* and *role-names*, respectively. $\mathcal{EL}$ *concept language* is defined by the grammar rule $C ::= A \mid \top \mid C \sqcap C \mid \exists r.C$, where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$.

The *semantics* of $\mathcal{EL}$ is given by an *interpretation*: that is a tuple $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a non-empty *domain* and $\cdot^{\mathcal{I}}$ is an *interpretation function* that maps every individual name a to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$; every concept name A to a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and every role name r to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The interpretation function $\cdot^{\mathcal{I}}$ is extended to $\mathcal{EL}$ concepts as shown in the upper part of Table 1.

The domain knowledge is encoded through a set of axioms, which restrict the interpretation domain of the concepts. A *TBox* $\mathcal{T}$ is a finite set of *general concept inclusions (GCIs)* of the form $C \sqsubseteq D$, where C, D are concepts. An *ABox* is a finite set of *concept assertions* $C(a)$ and *role assertions* $r(a,b)$, where $a, b \in \mathsf{N_I}$, C is a concept and $r \in \mathsf{N_R}$. A *knowledge base* is a pair $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ where $\mathcal{T}$ is a TBox and $\mathcal{A}$ is an ABox. We use the term *axiom* as a general expression for GCIs and assertions.

The interpretation $\mathcal{I}$ *satisfies* an axiom λ iff it satisfies the conditions on the lower part of Table 1. It is a *model* of the TBox $\mathcal{T}$ if it satisfies all GCIs in $\mathcal{T}$ and a *model* of the ABox $\mathcal{A}$ if it satisfies all the assertions in $\mathcal{A}$. An interpretation is a *model* of the KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ iff it is a model of both $\mathcal{T}$ and $\mathcal{A}$. For the rest of this paper we will denote as $\mathsf{N_I}(\mathcal{A})$ the set of all individual names that appear in the ABox $\mathcal{A}$.

CQA is an important reasoning task for DLs that has been investigated in the context of $\mathcal{EL}$. Let NV be a set of *variables* disjoint from $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_I}$. An *atom* is an expression of the form $A(\chi)$ or $r(\chi, \psi)$, where $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $\chi, \psi \in \mathsf{N_I} \cup \mathsf{NV}$. A *conjunctive query* (CQ) q is a non-empty set of atoms associated to a set $\mathsf{DV}(\mathsf{q}) \subseteq \mathsf{NV}$ of *distinguished variables*. If $\mathsf{DV}(\mathsf{q}) = \emptyset$, then q is called a *Boolean CQ*. A special case of a CQ is an *instance query (IQ)*, which consists of only one atom $A(\chi)$ with $A \in \mathsf{N_C}$.

Let q be a Boolean CQ and $\mathsf{IV}(\mathsf{q})$ be the set of all individual names and variables appearing in q. The interpretation $\mathcal{I}$ *satisfies* q if there exists a function $\pi : \mathsf{IV}(\mathsf{q}) \to \Delta^{\mathcal{I}}$ such that (i) $\pi(a) = a^{\mathcal{I}}$ for all $a \in \mathsf{N_I} \cap \mathsf{IV}(\mathsf{q})$, (ii) $\pi(\chi) \in A^{\mathcal{I}}$ for all $A(\chi) \in \mathsf{q}$, and (iii) $(\pi(\chi), \pi(\psi)) \in r^{\mathcal{I}}$ for all $r(\chi, \psi) \in \mathsf{q}$. In this case, we call π a *match* for $\mathcal{I}$ and q. The ontology $\mathcal{O}$ *entails* q ($\mathcal{O} \models \mathsf{q}$) iff every model of $\mathcal{O}$ satisfies q. For an arbitrary CQ q, a function $\mathfrak{a} : \mathsf{DV}(\mathsf{q}) \to \mathsf{N_I}(\mathcal{A})$ is an *answer* to q w.r.t. $\mathcal{O}$ iff $\mathcal{O}$ entails the Boolean CQ $\mathfrak{a}(\mathsf{q})$ obtained by replacing every distinguished variable $\chi \in \mathsf{DV}(\mathsf{q})$ with $\mathfrak{a}(\chi)$. *Conjunctive query answering* (CQA) is the task of finding all answers of a CQ, and query entailment is the problem of deciding whether an ontology entails a given Boolean CQ by replacing every distinguished variable $\chi \in \mathsf{DV}(\mathsf{q})$ with $\mathfrak{a}(\chi)$.

It is well known that query entailment in $\mathcal{EL}$ is polynomial w.r.t. data and KB complexity, but NP-complete w.r.t. combined complexity [23]. Notice that, $\mathcal{EL}$ does not enjoy the so-called full *first order rewritability* which has been considered as a key feature for CQA, since it allows one to reduce the problem to standard tasks in Relational Database Management Systems (RDMSs). Yet, CQA in $\mathcal{EL}$ can be successfully employed using a combined approach as described in [22].

3 The Bayesian Description Logic $\mathcal{BEL}$

The Bayesian DL $\mathcal{BEL}$ [7] has been introduced as a probabilistic extension of the light-weight DL $\mathcal{EL}$. In $\mathcal{BEL}$ probabilities are encoded through a *Bayesian network* (BN) [11]; that is, a pair $\mathcal{B} = (G, \Phi)$, where $G = (V, E)$ is a finite directed acyclic graph (DAG) whose nodes represent (boolean) random variables, and Φ contains, for every node $x \in V$, a conditional probability distribution $P_{\mathcal{B}}(x \mid \pi(x))$ of x given its parents $\pi(x)$. If V is the set of nodes in G, we say that $\mathcal{B}$ is a BN *over* V.

BNs are widely studied probabilistic graphical models where the underlying graph $G = (V, E)$ encodes a series of conditional independence assumptions between the random variables. Every variable $x \in V$ is known to be conditionally independent of its non-descendants given its parents. Thus, every BN $\mathcal{B}$ defines

a unique joint probability distribution (JPD) over V given by

$$P_{\mathcal{B}}(V) = \prod_{x \in V} P_{\mathcal{B}}(x \mid \pi(x)).$$

The concept language of $\mathcal{BEL}$ is the same as the $\mathcal{EL}$ concept language. The difference appears in encoding the domain knowledge, i.e. in forming axioms. $\mathcal{BEL}$ generalizes classical TBoxes (resp. ABoxes) by annotating the GCIs (resp. assertions) with a context defined by a set of literals belonging to a BN.

Formally, let $\mathsf{N_I}$ be a set of individual names and V a finite set of boolean variables. A *V-context* is a conjunction of literals over V. A *V-restricted general concept inclusion* (V-GCI) is an expression of the form $\langle C \sqsubseteq D : \kappa \rangle$ where C, D are $\mathcal{BEL}$ concepts and κ is a V-context. A *V-restricted assertion* (V-assertion) is an expression of the form $\langle C(a) : \kappa \rangle$, or $\langle r(a,b) : \kappa \rangle$ where $a, b \in \mathsf{N_I}$, C, D are $\mathcal{BEL}$ concepts and κ is a V-context. A *V-TBox (resp.V-ABox)* is a finite set of V-GCIs (resp.V-assertions). A $\mathcal{BEL}$ *knowledge base* (KB) is a tuple $\mathcal{K} = (\mathcal{B}, \mathcal{T}, \mathcal{A})$ where $\mathcal{B}$ is a BN over V, $\mathcal{T}$ is a V-TBox and $\mathcal{A}$ is a V-ABox.

We will sometimes speak of *contextual axioms* to address both V-GCIs and V-assertions. The intuition behind the contextual axioms is to enforce an axiom to hold within a given context, but not necessarily in others. The semantic of such axioms is realized with the so-called *contextual interpretations*, which differently from the classical interpretations also evaluate the context variables. Formally, given a finite set of Boolean variables V, $(\mathcal{I}, \mathcal{V}^{\mathcal{I}})$ is a *contextual interpretation* where $\mathcal{V}^{\mathcal{I}}$ is a *propositional interpretation* over V, and $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is a classical $\mathcal{EL}$ interpretation. We will usually ignore the prefix and speak simply of e.g. a *KB*, a *TBox*, an *ABox*, or an *interpretation*.

The interpretation function $\cdot^{\mathcal{I}}$ is extended to arbitrary $\mathcal{BEL}$ concepts as in $\mathcal{EL}$, i.e. using the rules in Table 1. We say that the contextual interpretation $(\mathcal{I}, \mathcal{V}^{\mathcal{I}})$ is a *model* of an axiom $\langle \lambda : \kappa \rangle$ denoted as $(\mathcal{I}, \mathcal{V}^{\mathcal{I}}) \models \langle \lambda : \kappa \rangle$, iff either (i) $\mathcal{V}^{\mathcal{I}} \not\models \kappa$, or (ii) $\mathcal{I} \models \lambda$. It is a *model* of the TBox $\mathcal{T}$ (resp. ABox $\mathcal{A}$) iff it is a model of all the axioms in $\mathcal{T}$ (resp. $\mathcal{A}$).

A contextual interpretation $(\mathcal{I}, \mathcal{V}^{\mathcal{I}})$ needs to satisfy only the axioms asserted within a context κ for which it holds that $\mathcal{V}^{\mathcal{I}} \models \kappa$. Formally, let $\mathcal{K} = (\mathcal{B}, \mathcal{T}, \mathcal{A})$ be a $\mathcal{BEL}$ KB: Given a contextual interpretation $(\mathcal{I}, \mathcal{V}^{\mathcal{I}})$ where $\mathcal{V}^{\mathcal{I}} = \mathcal{W}$, we define the $\mathcal{EL}$ KB $\mathcal{K}_{\mathcal{W}} = (\mathcal{T}_{\mathcal{W}}, \mathcal{A}_{\mathcal{W}})$ that needs to be satisfied by $\mathcal{I}$ as:

$$\mathcal{T}_{\mathcal{W}} := \{C \sqsubseteq D \mid \langle C \sqsubseteq D : \varphi \rangle \in \mathcal{T},\ \mathcal{W} \models \varphi\},$$
$$\mathcal{A}_{\mathcal{W}} := \{C(a) \mid \langle C(a) : \varphi \rangle \in \mathcal{A}, \mathcal{W} \models \varphi\} \cup \{r(a,b) \mid \langle r(a,b) : \varphi \rangle \in \mathcal{A}, \mathcal{W} \models \varphi\}.$$

In $\mathcal{BEL}$, uncertainty is represented through a BN that describes a joint probability distribution over the context variables. Semantically, $\mathcal{BEL}$ is linked to this distribution with the so called *multiple world semantics*: A *probabilistic interpretation* defines a probability distribution over a set of (contextual) interpretations; this distribution is required to be consistent with the joint probability distribution provided by the BN. Formally, a *probabilistic interpretation* is a pair $\mathcal{P} = (\mathfrak{I}, P_{\mathfrak{I}})$, where $\mathfrak{I}$ is a set of contextual interpretations and $P_{\mathfrak{I}}$ is a probability distribution over $\mathfrak{I}$ such that $P_{\mathfrak{I}}(\mathcal{I}, \mathcal{V}^{\mathcal{I}}) > 0$ only for finitely many interpretations

$(\mathcal{I}, \mathcal{V}^{\mathcal{I}}) \in \mathfrak{I}$. $\mathcal{P}$ is a *model* of the TBox $\mathcal{T}$ (resp. ABox $\mathcal{A}$) if every $(\mathcal{I}, \mathcal{V}^{\mathcal{I}}) \in \mathfrak{I}$ is a model of $\mathcal{T}$ (resp. $\mathcal{A}$). $\mathcal{P}$ is *consistent* with the BN $\mathcal{B}$ if for every possible valuation $\mathcal{W}$ of the variables in V it holds that

$$\sum_{(\mathcal{I}, \mathcal{V}^{\mathcal{I}}) \in \mathfrak{I},\, \mathcal{V}^{\mathcal{I}} = \mathcal{W}} P_{\mathfrak{I}}(\mathcal{I}, \mathcal{V}^{\mathcal{I}}) = P_{\mathcal{B}}(\mathcal{W}).$$

The probabilistic interpretation $\mathcal{P}$ is a *model* of the KB $(\mathcal{B}, \mathcal{T}, \mathcal{A})$ iff it is a (probabilistic) model of $\mathcal{T}$, $\mathcal{A}$ and consistent with $\mathcal{B}$.

To provide a fine-grained analysis of the complexity of reasoning in $\mathcal{BEL}$, we use different measures for the size of the input. In *data complexity*, we measure only the size of the ABox, and consider the rest of the KB and the query fixed. For *ontology complexity* we use the size of the TBox and the ABox; in *network complexity* the relevant input is the BN, while the *combined complexity* considers the size of the whole input.

4 Probabilistic Query Entailment

Different reasoning tasks have been studied in the context of Bayesian DLs; perhaps the most prominent one being the *probabilistic entailment* [6]. Although, probabilistic entailment has been considered generally, its focus was on entailments of simple consequences, i.e. consequences of the form subsumption, instance checking etc., all of which are tasks that can be decided in time polynomial in $\mathcal{EL}$. Thus, the class of problems based on entailments of simple consequences has lead tight complexity bounds in $\mathcal{BEL}$ [8].

Here we generalize these results and study *probabilistic query entailment*. In this setting, we are not just interested in the entailment of a query q but also in the probability of such entailment.

Definition 1 (probabilistic query entailment). *Let $\mathcal{K} = (\mathcal{B}, \mathcal{T}, \mathcal{A})$ be a $\mathcal{BEL}$ KB over V and $\mathcal{P} = (\mathfrak{I}, P)$ a probabilistic interpretation. $\mathcal{P}$ defines a probability distribution $P_{\mathcal{P}}$ over all conjunctive queries q given by*

$$P_{\mathcal{P}}(\mathsf{q}) := \sum_{(\mathcal{I}, \mathcal{V}^{\mathcal{I}}) \in \mathfrak{I},\ \mathcal{I} \models \mathsf{q}} P(\mathcal{I}, \mathcal{V}^{\mathcal{I}}).$$

The probability of the query q *w.r.t.* $\mathcal{K}$ *is* $P_{\mathcal{K}}(\mathsf{q}) := \inf_{\mathcal{P} \models \mathcal{K}} P_{\mathcal{P}}(\mathsf{q})$. *A query* q *is entailed with probability* $p \in (0, 1]$ *iff* $P_{\mathcal{K}}(\mathsf{q}) \geq p$.

Recall that every valuation $\mathcal{W}$ defines an $\mathcal{EL}$ ontology that contains all the axioms that must be satisfied by any contextual interpretation using the valuation $\mathcal{W}$. Given a Boolean CQ q, we can build a probabilistic model $\mathcal{P}_{\mathsf{q}} = (\mathfrak{I}, P)$ of $\mathcal{K}$ such that for every valuation $\mathcal{W}$ there is exactly one contextual interpretation $\mathcal{I}_{\mathcal{W}} \in \mathfrak{I}$, and it satisfies that $\mathcal{I}_{\mathcal{W}} \models \mathsf{q}$ iff $\mathcal{K}_{\mathcal{W}} \models \mathsf{q}$. It is easy to see that every other model $\mathcal{P}$ of $\mathcal{K}$ is such that $P_{\mathcal{P}}(\mathsf{q}) \geq P_{\mathcal{P}_{\mathsf{q}}}(\mathsf{q})$, which yields the following theorem.

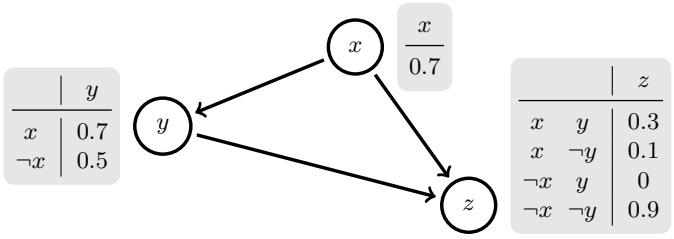

Fig. 1: The BN $\mathcal{B}_{\texttt{ABC}}$ over the variables $\{x, y, z\}$

Theorem 2. *For every $\mathcal{BEL}$ KB $\mathcal{K}$ and Boolean CQ $\mathtt{q}$ $P_{\mathcal{K}}(\mathtt{q}) = \sum_{\mathcal{K}_{\mathcal{W}} \models \mathtt{q}} P_{\mathcal{B}}(\mathcal{W})$.*

Given Theorem 2, one can compute the probability of any query by summing up the probabilities of the worlds that entail the query $\mathtt{q}$. We illustrate probabilistic query entailment with a simple example.

Example 3. Consider the $\mathcal{BEL}$ KB $\mathcal{K} = ((\mathcal{T}_{\texttt{ABC}}, \mathcal{A}_{\texttt{ABC}}), \mathcal{B}_{\texttt{ABC}})$ where

$$\mathcal{T}_{\texttt{ABC}} := \{ \langle \mathtt{A} \sqsubseteq \exists \mathtt{r}.\mathtt{B} : \{y\} \rangle , \langle \mathtt{B} \sqsubseteq \mathtt{C} : \{x\} \rangle \}$$
$$\mathcal{A}_{\texttt{ABC}} := \{ \langle \mathtt{A}(\mathtt{a}) : \{\mathtt{x}\} \rangle , \langle \mathtt{r}(\mathtt{a}, \mathtt{b}) : \{\mathtt{z}\} \rangle , \langle \mathtt{C}(\mathtt{b}) : \{x, z\} \rangle , \langle \mathtt{A}(\mathtt{c}) : \{\mathtt{y}\} \rangle \}$$

$\mathcal{B}_{\texttt{ABC}}$ is the BN given in Figure 1 and the Boolean CQ $\mathtt{q} = \{A(\chi), r(\chi, \psi), C(\psi)\}$. Clearly, $\mathcal{K}_{\mathcal{W}} \models \mathtt{q}$ only for worlds $\mathcal{W}$ such that $\mathcal{W} \models (x \wedge y) \vee (x \wedge z)$. Hence, we get $P_{\mathcal{K}}(\mathtt{q}) = P_{\mathcal{B}_{\texttt{ABC}}}((x \wedge y) \vee (x \wedge z)) = 0.411$.

Clearly the number of worlds might be exponential in $|V|$. In fact, this corresponds to exponentially many query entailment tests, which can be performed using polynomial space only.

Theorem 4. *Probabilistic query entailment is polynomial w.r.t. data and ontology complexity; and in* PSPACE *w.r.t. network and combined complexity.*

The bounds for network and combined complexity can be improved if we restrict the queries to instance queries only. It is then possible to use a novel structure, called the *proof structure* such as the one presented in [8]. The general idea is to reduce probabilistic reasoning in $\mathcal{BEL}$ knowledge bases to standard inferences in a BN. In essence, a proof structure compactly describes the class of contexts that entail the wanted consequence. Using this proof-structure, it is possible to construct a BN from which the probability of such consequence can be computed. Importantly, it has been shown that such reduction can be performed in polynomial time.

In a nutshell, a proof structure is a directed acyclic hyper-graph, in which every node represents an axiom. It is constructed in a bottom up manner with the help of a set of deduction rules. Starting from an initial set of axioms given by the KB, it adds new nodes for the axioms resulting from 1-step application of the deduction rules. Edges are used for denoting the axioms that have been used for the deduction. This process continues until the rules are saturated under

the set of axioms. This structure enables us to trace back all the causes for a consequence. Thus, once transformed into a BN, it represents all contexts for a consequence, the probability of which can then be computed via the BN. For the details, we refer to [8].

To provide a better complexity bound for probabilistic query entailment, we extend the proof structure to also handle the assertional knowledge, which was not present so far. Following a naïve approach it is possible to introduce a new set of deduction rules; instead, we make use of nominals to handle the assertions.

Briefly, the DL $\mathcal{ELO}$ extends $\mathcal{EL}$ with nominals; that is, it allows special types of concepts of the form $\{a\}$ with the semantics $\{a^{\mathcal{I}}\}$. It is well-known that in the presence of nominals, $\mathcal{EL}$ KBs can be represented without an ABox. Thus, for an $\mathcal{ELO}$ KB it is possible to benefit from the deduction rules presented in [19] to construct a proof structure. Using the approach in [8] with the new rules given in [19] over an $\mathcal{ELO}$ KB, we construct a proof structure for an $\mathcal{EL}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ that is guaranteed to contain the information of all possible causes for a consequence to follow from $\mathcal{K}$. Moreover, this hypergraph is acyclic and has polynomially many nodes, on the size of $\mathcal{K}$, by the properties of the rules and their applications.

A $\mathcal{BEL}$ KB can be transformed into a $\mathcal{BELO}$ KB in the obvious way. Let $\mathcal{K} = \{\mathcal{B}, \mathcal{T}, \mathcal{A}\}$ be a $\mathcal{BEL}$ KB, we construct the $\mathcal{BELO}$ KB $\mathcal{K}' = \{\mathcal{B}, \mathcal{T}'\}$ where

$$\begin{aligned}\mathcal{T}' \;=\; \mathcal{T} \;\cup\; \{\; \langle \{a\} \sqsubseteq C : \kappa \rangle \;\mid\; \langle C(a) : \kappa \rangle \in \mathcal{A} \;\} \\ \cup\; \{\; \langle \{a\} \sqsubseteq \exists . r\{b\} : \kappa \rangle \;\mid\; \langle r(a,b) : \kappa \rangle \in \mathcal{A} \;\}.\end{aligned}$$

Clearly, $\mathcal{K} \models c$ iff $\mathcal{K}' \models c$ for any consequence c. Hence, for any ABox assertion, it is possible to construct a proof structure of polynomial size. To check the probability of an instance query $C(\chi)$ we construct a BN using the proof structures of $C(a)$ where a is an individual appearing in the ABox. Observe that the number of proof structures is bound with the individuals available in the ABox and we obtain a polynomial construction w.r.t. the size of the input.

Together with the hardness of probabilistic entailment of simple consequences in $\mathcal{BEL}$ without ABoxes, we get the following result.

Lemma 5. *Probabilistic query entailment restricted to IQs is* PP*-complete w.r.t. the combined complexity.*

5 Probabilistic Query Answering

Query answering is the problem of finding mappings for a query, i.e. one is not just interested whether a query is entailed or not, but also with the witnesses of such entailment. Typically, data is assumed to be large and it is not always very feasible to return all answers to a query q to the user. One of the most important applications of query answering is returning the top-k answers to a given query q w.r.t. a measure. By this way users do not only get a feasible number of answers but also a fine grained view over the data. In the context of probabilities, we are interested in finding the answers that are most likely.

Let q be a query with the distinguished variables $\mathsf{DV}(\mathsf{q})$, and $\mathcal{K} = (\mathcal{B}, \mathcal{T}, \mathcal{A})$ a $\mathcal{BEL}$ KB. We denote by $\mathsf{Ind}(\mathcal{A})$ the set of all individual names appearing in $\mathcal{A}$. Recall that every function $\mathfrak{a} : \mathsf{DV}(\mathsf{q}) \to \mathsf{Ind}(\mathcal{A})$ defines a CQ obtained by replacing every $\chi \in \mathsf{DV}(\mathsf{q})$ in q with $\mathfrak{a}(\chi)$. Abusing the notation, we call this query $\mathfrak{a}(\mathsf{q})$. We call any function $\mathfrak{a} : \mathsf{DV}(\mathsf{q}) \to \mathsf{Ind}(\mathcal{A})$ an *answer* to q w.r.t. $\mathcal{K}$, and define its probability as $P_{\mathcal{K}}(\mathfrak{a}) := P_{\mathcal{K}}(\mathfrak{a}(\mathsf{q}))$. Since an answer defines a boolean CQ, all complexity results for CQs transfer immediately. Every answer to a query q, has a probability, which we use as a measure to distinguish the answers. We refine the set of answers w.r.t. their probabilities and return top answers only.

Definition 6 (top-k answer). *Let q be a query, $\mathcal{K}$ be a $\mathcal{BEL}$ KB, and $k \in \mathbb{N}$. A* top-k answer *to q w.r.t. $\mathcal{K}$ is a tuple $(\mathfrak{a}_1, \ldots, \mathfrak{a}_k)$ of different answers to q w.r.t. $\mathcal{K}$ such that (i) for all $i, 1 \leq i < k$, $P_{\mathcal{K}}(\mathfrak{a}_i) \geq P_{\mathcal{K}}(\mathfrak{a}_{i+1})$, and (ii) for every other answer $\mathfrak{a}$, $P_{\mathcal{K}}(\mathfrak{a}_k) \geq P_{\mathcal{K}}(\mathfrak{a})$.*

In other words, a top-k answer is an ordered tuple of the k answers with the highest probability. We assume that k is a constant that is fixed *a priori*. Thus, it is not considered part of the input of the problem. Obviously, since different answers may have the same probability, top-k answers are not unique. Here we are only interested in finding one of them. Stating it as a decision problem, we want to verify whether a given tuple is a top-k answer.

Example 7. Consider the $\mathcal{BEL}$ KB $\mathcal{K} = ((\mathcal{T}_{\texttt{ABC}}, \mathcal{A}_{\texttt{ABC}}), \mathcal{B}_{\texttt{ABC}})$ provided in Example 3 and the query $\mathsf{q} = \{A(\chi)\}$ with $\chi \in \mathsf{DV}$. We are interested in identifying the top-1 answer to q w.r.t. $\mathcal{K}$. Notice that both $\mathfrak{a}_0 : \chi \mapsto a$ and $\mathfrak{a}_1 : \chi \mapsto c$ are answers to q with positive probability. Clearly, $\mathfrak{a}_0$ is the top-1 answer since $P_{\mathcal{K}}(\mathfrak{a}_0) > P_{\mathcal{K}}(\mathfrak{a}_1)$.

Assuming that the size of q and the BN $\mathcal{B}$ are fixed, there are polynomially many answers to q w.r.t. $\mathcal{K}$, and for each answer $\mathfrak{a}$, we can compute $P_{\mathcal{B}}(\mathfrak{a})$ performing polynomially many $\mathcal{EL}$ query entailment tests. Thus, it is possible to verify whether $(\mathfrak{a}_1, \ldots, \mathfrak{a}_k)$ is a top-k answer in polynomial time w.r.t. ontology complexity.

If we consider the combined complexity, the problem can be decided as follows. For every answer to the query, we only keep track of those answers that are best by checking the probabilities $P_{\mathcal{B}}(\mathfrak{a})$ of the individual answers iteratively. Since the latter can be done in PSPACE, we obtain an upper bound.

Theorem 8. *Let $\mathfrak{A} = (\mathfrak{a}_1, \ldots, \mathfrak{a}_k)$ be a tuple of answers to q w.r.t. $\mathcal{K}$. Deciding whether $\mathfrak{A}$ is a top-k answer is polynomial w.r.t. data and ontology complexity, in* PSPACE *w.r.t. network complexity and combined complexity.*

We show a lower bound for this problem w.r.t. the combined complexity. by providing a reduction from the decision version of the *maximum a-posteriori* (D-MAP) problem for BNs [11]. Formally, given a BN $\mathcal{B}$ over V, a set $Q \subseteq V$, a context κ, and $p > 0$, the D-MAP problem consists of deciding whether there exists a valuation μ of the variables in Q such that $P_{\mathcal{B}}(\kappa \wedge \mu) > p$.

Consider an arbitrary but fixed instance of D-MAP described by the BN $\mathcal{B} = ((V, E), \Phi)$, the context κ, $Q \subseteq V$, and $p > 0$. We introduce a new Boolean random variable z not appearing in V. Using this variable, we construct a new DAG (V', E) with $V' = V \cup \{z\}$ and a new BN $\mathcal{B}' = ((V', E), \Phi')$, where $P_{\mathcal{B}'}(v \mid \pi(v)) = P_{\mathcal{B}}(v \mid \pi(x))$ for all $v \in V$, and $P_{\mathcal{B}'}(z) = p$. Consider the $\mathcal{BEL}$ KB $\mathcal{K} = (\mathcal{B}', \emptyset, \mathcal{A})$ where

$$\mathcal{A} := \{\langle A_x(a_x) : x\rangle, \langle A_x(b_x) : \neg x\rangle, \langle A_x(c) : z\rangle \mid x \in Q\} \cup \{\langle B(a) : \kappa\rangle, \langle B(c) : z\rangle\},$$

and query $\mathfrak{q} := \{A_x(\chi_x) \mid x \in Q\} \cup \{B(\chi)\}$, where all the variables are distinguished; i.e., $\mathsf{DV}(\mathfrak{q}) = \{\chi_x \mid x \in Q\} \cup \{\chi\}$. It is easy to see that the mapping $\mathfrak{a}_0 : \mathsf{DV}(\mathfrak{q}) \to \{c\}$ is an answer to this query and $P_{\mathcal{K}}(\mathfrak{a}_0) = p$. Moreover, any other answer that maps any variable to c will have the probability at most p, since it can only be entailed in contexts satisfying z. Suppose that there is an answer $\mathfrak{a}$ such that $P_{\mathcal{K}}(\mathfrak{a}) > p$. This answer must map every variable χ_x to either a_x or b_x and χ to a. Let $\mu_{\mathfrak{a}} := \bigwedge_{\mathfrak{a}(\chi_x)=a_x} x \wedge \bigwedge_{\mathfrak{a}(\chi_x)=b_x} \neg x$. By construction, $\mu_{\mathfrak{a}}$ is a valuation of the variables in Q, $P_{\mathcal{B}}(\kappa \wedge \mu_{\mathfrak{a}}) > p$, and $\mathfrak{a}(\mathfrak{q})$ is only entailed by valuations satisfying the context $\kappa \wedge \mu_{\mathfrak{a}}$. Overall this means that $\mathfrak{a}_0$ is *not* a top-1 answer iff there is a valuation μ of the variables in Q such that $P_{\mathcal{B}}(\kappa \wedge \mu) > p$.

Theorem 9. *Deciding whether a tuple $\mathfrak{A}$ is a top-k answer is co*$\mathrm{NP}^{\mathrm{PP}}$*-hard w.r.t. combined complexity.*

Notice that the proof uses a very simple query which is in fact acyclic. Thus, contrary to classical $\mathcal{EL}$ [4], restricting to acyclic queries does not suffice for reducing the complexity of reasoning. Clearly, if we consider IQs this hardness might not hold any more.

Obtaining most probable answers for a query is a crucial task for the domains where imprecise characterizations of knowledge is necessary. The next section is dedicated to another reasoning task that can be seen dual to top-k answers, namely top-k contexts.

6 Most Likely Contexts for a Query

Dually to finding the most likely answers to a query, we are also interested in finding the k most likely contexts that entail a given Boolean query $\mathfrak{q}$. More precisely, suppose that we have already observed that the query $\mathfrak{q}$ holds; then, we are interested in finding out which is the current context. As in the previous section, we do not consider one, but search for a fixed number of contexts that are the most likely to hold.

To define this reasoning task formally, we must generalize the notion of the ontology $\mathcal{K}_{\mathcal{W}}$ defined to consider arbitrary contexts κ, which we denote as $\mathcal{K}_\kappa$. For any contextual interpretation $(\mathcal{I}, \mathcal{V}^{\mathcal{I}})$ with $\mathcal{V}^{\mathcal{I}} \models \kappa$ it must hold that $\mathcal{I} \models \mathcal{K}_\kappa$. If $\mathcal{K}_\kappa$ entails the Boolean query $\mathfrak{q}$, then we say that $\mathfrak{q}$ holds in context κ. We are interested in finding out the most likely contexts in which a given query holds.

Definition 10 (top-k contexts). *Let* q *be a CQ,* $\mathcal{K}$ *a* $\mathcal{BEL}$ *KB, and* $k \in \mathbb{N}$. $\kappa_1, \ldots, \kappa_k$ *are* top-k contexts *for* q *w.r.t.* $\mathcal{K}$ *if* $\mathcal{K}_{\kappa_i}$ *entails* q *for all* $i, 1 \leq i \leq k$*;* $P_{\mathcal{B}}(\kappa_i) \geq P_{\mathcal{B}}(\kappa_{i+1})$ *for all* $i, 1 \leq i \leq k$*; and there is no other context* κ *such that* $\mathcal{K}_{\kappa} \models \mathsf{q}$ *and* $P_{\mathcal{B}}(\kappa) > P_{\mathcal{B}}(\kappa_k)$.

We illustrate top-k mlc with our continuing example. In this case, we are interested in finding out the 2 most likely context that entail the query.

Example 11. Consider the $\mathcal{BEL}$ KB $\mathcal{K} = ((\mathcal{T}_{\mathtt{ABC}}, \mathcal{A}_{\mathtt{ABC}}), \mathcal{B}_{\mathtt{ABC}})$ and query q provided in Example 3. Clearly all contexts κ that entail q are such that $\kappa \models \{x, y\} \vee \{x, z\}$. The top-2 contexts are then $\langle \{x, y\}, \{x, z\} \rangle$ since $P_{\mathcal{B}_{\mathtt{ABC}}}(\{x, y\}) > P_{\mathcal{B}_{\mathtt{ABC}}}(\{x, z\})$.

The problem of finding one most likely context has been studied for simple queries. In those special cases, it was shown to be $\mathrm{coNP}^{\mathrm{PP}}$-complete problem w.r.t. combined complexity [8]. The $\mathrm{coNP}^{\mathrm{PP}}$ upper bound holds also for top-k contexts w.r.t. combined complexity: if a tuple is not a top-k mlc, then guess a new context κ and show using a PP oracle that $\mathcal{K}_{\kappa} \models \mathsf{q}$ and $P_{\mathcal{B}}(\kappa) > P_{\mathcal{B}}(\kappa_k)$. If the BN is fixed, then the number of contexts is constant, and they can be ordered w.r.t. their complexity in constant time. The top-k mlc problem is then solved by applying a constant number of $\mathcal{EL}$ CQ entailment tests, yielding a polynomial upper bound w.r.t. ontology complexity. All these complexity results are summarized in the following theorem.

Theorem 12. *Deciding whether* $\kappa_1, \ldots, \kappa_k$ *are top-k mlc for* q *w.r.t. the KB* $\mathcal{K}$ *is polynomial w.r.t. data, and ontology complexity,* PP*-hard and in* $\mathrm{NP}^{\mathrm{PP}}$ *w.r.t. network complexity, and* $\mathrm{NP}^{\mathrm{PP}}$*-complete w.r.t. combined complexity.*

Given the hardness of deciding top-k contexts, we consider a special case of this problem: Suppose now that all contexts are of a special form, i.e. they are valuations, we call this problem *top-k worlds*. In this case, we need to guess a world $\mathcal{W}$ and decide whether i) $\mathcal{K}_{\mathcal{W}} \models q$ and ii) $P_{\mathcal{K}}(\mathcal{W}) > P_{\mathcal{K}}(\mathcal{W}_k)$, where the former requires an NP oracle whereas the latter can be decided in polynomial time using tha standard chain rule of BNs.

Notice that, top-k contexts and top-k answers are dual to each other, but they do not necessarily overlap. Consider for instance the case, where all top-k answers to a query q are retrieved from the same context κ. In this case, top-k contexts for q will contain other contexts than κ with the assumption that $k > 1$. Deciding top-k contexts is particularly informative for cases where the diversity of knowledge is important.

We have discussed several reasoning problems in $\mathcal{BEL}$ w.r.t. CQs which we considered as natural problems that could arise in several domains. For a summary of the results, see Table 2.

7 Related Work

The literature on probabilistic extensions of DLs consists of various formalisms, each of which with different characteristics [20]. Despite the fact that probabilistic query answering has been studied widely in relational databases [15, 13, 9],

Table 2: $\mathcal{BEL}$ reasoning problems and their complexity

Problem	**data**	**ontology**	**network**	**combined**
probabilistic CQ entailment	P	P	PP-c	PP/PSPACE
probabilistic IQ entailment	P	P	PP-c	PP-c.
top-k answer	P	P	PP/PSPACE	coNP^{PP}/PSPACE
top-k contexts	P	P	$\text{PP}/\text{coNP}^{\text{PP}}$	coNP^{PP}/PSPACE
top-k worlds	P	P	coNP-c	coNP/Π_2^p

RDF graphs [16] and XML databases [1, 17], only few of the probabilistic DLs considered CQA as a reasoning task.

In the probabilistic extension of Datalog+/- [14] authors are interested in retrieving the answers that are above a threshold value that is set *a priori*. In contrast to $\mathcal{BEL}$, in probabilistic Datalog+/- the underlying semantics is based on Markov logic networks. The Prob-DL family [21] extends classical DLs with subjective probabilities, also known as Type II probabilities [18]. The main difference with our logic is that Prob-$\mathcal{EL}$ introduces probabilities as a concept constructor, whereas we allow only probabilities over axioms. More closely related to $\mathcal{BEL}$ is BDL-Lite [10]. As is in $\mathcal{BEL}$, BDL-Lite only allows probabilities over axioms and conditional dependencies are represented faithfully. However, as it has been pointed before [8], the authors use a closed world assumption, which easily leads to inconsistencies for the Bayesian extension of $\mathcal{EL}$.

8 Conclusions

We have studied probabilistic query entailment, top-k answers and top-k contexts as reasoning problems. Though not being complete, for each of these problems, we provided a complexity analysis. Moreover, we have shown that assuming that the given BN and query are relatively small, all problems become tractable. Removing this assumption immediately results in the loss of tractability, which is not surprising given the intractability results in BNs and CQA in $\mathcal{EL}$.

As a future work, we want to obtain tight bounds w.r.t. all measures provided. We have shown tight complexity bounds for the query entailment problem of IQs. Restricting our attention to IQs, other problems might also get easier under widely accepted assumptions of complexity theory. It should be reminded that this is unfortunately not the case for acyclic queries.

On the practical side, we will consider optimizing the reasoning mechanisms and we will implement a system for reasoning in $\mathcal{BEL}$ that will benefit both from techniques in DLs, such as module extraction, query processing and from techniques in reasoning with lifted BNs, mainly based on logic programming as in [12].

References

1. Abiteboul, S., Senellart, P.: Querying and updating probabilistic information in XML. In: Proc. of EDBT'06. LNCS, vol. 3896. Springer Verlag (2006)
2. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Proc. of IJCAI'05. Morgan Kaufmann Publishers (2005)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, 2nd edn. (2007)
4. Bienvenu, M., Ortiz, M., Šimkus, M., Xiao, G.: Tractable queries for lightweight description logics. In: Proc. of IJCAI'13. AAAI Press (2013)
5. Brandt, S.: Polynomial Time Reasoning in a Description Logic with Existential Restrictions, GCI Axioms, and—What Else? In: Proc. of ECAI'04. vol. 110. IOS Press (2004)
6. Ceylan, İ.İ., Peñaloza, R.: Bayesian Description Logics. In: Proc. of DL'14. CEUR Workshop Proceedings, vol. 1193. CEUR-WS (2014)
7. Ceylan, İ.İ., Peñaloza, R.: The Bayesian Description Logic $\mathcal{BEL}$. In: Proc. of IJCAR'14. LNCS, vol. 8562. Springer Verlag (2014)
8. Ceylan, İ.İ., Peñaloza, R.: Tight Complexity Bounds for Reasoning in the Description Logic $\mathcal{BEL}$. In: Proc. of JELIA'14. LNCS, vol. 8761. Springer Verlag (2014)
9. Dalvi, N., Suciu, D.: Efficient query evaluation on probabilistic databases. VLDB Journal 16(4) (2007)
10. D'Amato, C., Fanizzi, N., Lukasiewicz, T.: Tractable Reasoning with Bayesian Description Logics. In: Proc. of SUM'08. LNCS, vol. 5291. Springer Verlag (2008)
11. Darwiche, A.: Modeling and Reasoning with Bayesian Networks. Cambridge University Press (2009)
12. De Raedt, L., Kimmig, A., Toivonen, H.: ProbLog: A probabilistic prolog and its application in link discovery. In: Proc. of IJCAI'07. Morgan-Kaufmann Pub. (2007)
13. Fuhr, N., Rölleke, T.: A probabilistic relational algebra for the integration of information retrieval and database systems. ACM TOIS'97 15(1) (1997)
14. Gottlob, G., Lukasiewicz, T., Martinez, M.V., Simari, G.I.: Query answering under probabilistic uncertainty in datalog +/- ontologies. Ann. Math. AI 69(1) (2013)
15. Grädel, E., Gurevich, Y., Hirsch, C.: The complexity of query reliability. In: Proc. ACM SIGACT-SIGMOD-SIGART'98 (1998)
16. Huang, H., Liu, C.: Query evaluation on probabilistic RDF databases. In: WISE09,. LNCS, vol. 5802. Springer Verlag (2009)
17. Hung, E., Getoor, L., Subrahmanian, V.S.: PXML: A probabilistic semistructured data model and algebra. In: Proc. ICDE'03 (2003)
18. Joseph, H.: An Analysis of First - Order Logics of Probability. In: Proc. of IJCAI'89. Morgan Kaufmann Publishers (1989)
19. Kazakov, Y., Krötzsch, M.R., Simančík, F.: Practical Reasoning with Nominals in the EL Family of Description Logics. In: Proc. of KR'12. AAAI Press (2012)
20. Lukasiewicz, T., Straccia, U.: Managing uncertainty and vagueness in description logics for the Semantic Web. Web Semantics: Science, Services and Agents on the World Wide Web 6(4), 291–308 (Nov 2008)
21. Lutz, C., Schröder, L.: Probabilistic Description Logics for Subjective Uncertainty. In: Proc. of KR'10. AAAI Press (2010)
22. Lutz, C., Toman, D., Wolter, F.: Conjunctive Query Answering in the Description Logic $\mathcal{EL}$ Using a Relational Database System. In: Proc. of IJCAI'09. AAAI (2009)
23. Rosati, R.: On conjunctive query answering in EL. In: Proc. of DL'07. CEUR Workshop Proceedings, vol. 250. CEUR-WS (2007)