Ontology-Based Query Answering for Probabilistic Temporal Data

Patrick Koopmann
Institute for Theoretical Computer Science
Technische Universität Dresden, Germany
firstname.lastname@tu-dresden.de

Abstract

We investigate ontology-based query answering for data that are both temporal and probabilistic, which might occur in contexts such as stream reasoning or situation recognition with uncertain data. We present a framework that allows to represent temporal probabilistic data, and introduce a query language with which complex temporal and probabilistic patterns can be described. Specifically, this language combines conjunctive queries with operators from linear time logic as well as probability operators. We analyse the complexities of evaluating queries in this language in various settings. While in some cases, combining the temporal and the probabilistic dimension in such a way comes at the cost of increased complexity, we also determine cases for which this increase can be avoided.

Introduction

The internet has become highly dynamic, with information being frequently added and changed, and new data being generated from a variety of sources. In addition, new technologies such as smart phones and the internet of things (IoT) frequently encounter a data environment that is constantly changing. To make use of these data, there has been an increasing interest in investigating semantic and reasoning techniques that process not only static data, but streams of data, such as in the semantic stream reasoning paradigm (Margara et al. 2014). One application is that of *situation recognition*, where we want to recognise or query temporal patterns in a stream of data. As Margara et al. illustrate, frequently, the data encountered in stream reasoning applications is not only temporal, but also probabilistic in nature. In *ontology-based query answering* (OBQA), queries are evaluated with respect to an *ontology*, which specifies background knowledge about the domain of interest. Using a reasoner, this allows to query also information that follows implicitly from the data. While OBQA was originally designed for querying static and precise data, there is good motivation also for semantic stream reasoning as well as for querying historical data, where data are temporal and probabilistic.

As an example, consider a health or fitness monitoring application, for which one may want to use concepts from a medical ontology such as SNOMED CT (Elkin et al. 2006) to describe information about the health status of a patient. Specifically, such an application could be used on a smartphone in combination with a sensor that measures the diastolic blood pressure of the patient while he is exercising (Kumar et al. 2015). As the sensor might be imprecise in its measurements, it might report information about whether the blood pressure of the patient is high with an associated probability, and provide this information to the application in regular time intervals. If a too high blood pressure was observed for several times during a short period, the application should give a warning to the patient, and advise him to take a break from his exercise.

We assume a representation of the data in form of a sequence of probabilistic data sets, which may have been obtained using further preprocessing and windowing operations. A typical query would then ask whether, with a high probability, the patient had at least twice a high blood pressure during the last 10 minutes. In order to properly take both the temporal and the probabilistic aspects of this question into account when querying the probabilistic stream, we propose a query language for OBQA that comes with both temporal and probabilistic operators. In this language, the query would be expressed as follows, where $\mathsf{HighBloodPressure}$ is a concept defined in the ontology.

$$\mathsf{P}_{>.8}(\bigcirc^{-10}\Diamond(\mathsf{HighBloodPressure}(x) \\ \wedge \bigcirc\Diamond\mathsf{HighBloodPressure}(x)))$$

Our language is an extension of the well-investigated temporal query language introduced in (Borgwardt and Thost 2015; Baader, Borgwardt, and Lippmann 2015),which extends conjunctive queries with operators from linear temporal logic (LTL). Other authors considered using these operators also as part of the DL, either to describe temporal concepts (Gabbay et al. 2003), or to make the axioms of the ontology itself temporal (Baader, Ghilardi, and Lutz 2012). Recently, this work has been extended also to metric temporal logics (Baader et al. 2017; Brandt et al. 2017). Temporal reasoning for streams of data has also been considered in

Copyright © 2019, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

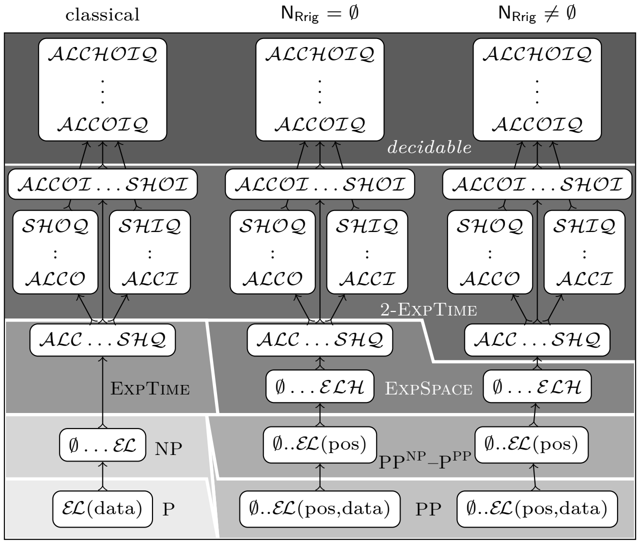
classical
$\mathsf{N}_{\mathsf{Rrig}} = \emptyset$
$\mathsf{N}_{\mathsf{Rrig}} \neq \emptyset$
$\mathcal{ALCHOIQ}$
$\vdots$
$\mathcal{ALCOIQ}$
$\mathcal{ALCHOIQ}$
$\vdots$
$\mathcal{ALCOIQ}$
$\mathcal{ALCHOIQ}$
$\vdots$
$\mathcal{ALCOIQ}$
decidable
$\mathcal{ALCOI} \dots \mathcal{SHOI}$
$\mathcal{ALCOI} \dots \mathcal{SHOI}$
$\mathcal{ALCOI} \dots \mathcal{SHOI}$
$\mathcal{SHOQ}$
:
$\mathcal{ALCO}$
$\mathcal{SHIQ}$
:
$\mathcal{ALCI}$
$\mathcal{SHOQ}$
:
$\mathcal{ALCO}$
$\mathcal{SHIQ}$
:
$\mathcal{ALCI}$
$\mathcal{SHOQ}$
:
$\mathcal{ALCO}$
$\mathcal{SHIQ}$
:
$\mathcal{ALCI}$
2-ExpTime
$\mathcal{ALC} \dots \mathcal{SHQ}$
$\mathcal{ALC} \dots \mathcal{SHQ}$
$\mathcal{ALC} \dots \mathcal{SHQ}$
ExpTime
$\emptyset \dots \mathcal{ELH}$
ExpSpace
$\emptyset \dots \mathcal{ELH}$
$\emptyset \dots \mathcal{EL}$
NP
$\emptyset..\mathcal{EL}$(pos)
$\mathrm{PP}^{\mathrm{NP}}$–$\mathrm{P}^{\mathrm{PP}}$
$\emptyset..\mathcal{EL}$(pos)
$\mathcal{EL}$(data)
P
$\emptyset..\mathcal{EL}$(pos,data)
PP
$\emptyset..\mathcal{EL}$(pos,data)

Figure 1: Complexity of TPQ Entailment vs. classical CQ entailment. Here, $\emptyset$ corresponds to the case without TBox. Except for the P^{PP} and the decidability results, all complexity bounds are tight.

the context of Datalog (Ronca et al. 2018). A recent survey on temporal query answering with ontologies can be found in (Artale et al. 2017).

A major restriction of using temporal concepts in the DL is that we cannot keep relations between objects stable throughout the timeline (rigid) without making the DL undecidable. This limits their application for querying situations concerning more than one object, which in applications more involved than in our running example might be crucial. For this reason, we focus on extensions of the query language rather than the DL in this paper, though investigating an extension of this framework with temporal DLs might be interesting as future work.

In addition to the temporal dimension, we add a probabilistic dimension to our setting. An OBQA framework for *probabilistic* data was presented in (Jung and Lutz 2012), though a temporal dimension was not considered here yet. Since this publication, several authors investigated OBQA in similar settings (Borgwardt, Ceylan, and Lukasiewicz 2017; Baader, Koopmann, and Turhan 2017; Ceylan and Peñaloza 2017). In addition to settings based on probabilistic databases, there is also research on extending DLs with probability operators, such as in $\mathbf{P}$-$\mathcal{SHIF}(\mathcal{D})$/$\mathbf{P}$-$\mathcal{SHOIN}(\mathcal{D})$ (Lukasiewicz 2008) or Prob-$\mathcal{ALC}$/Prob-$\mathcal{EL}$ (Gutiérrez-Basulto et al. 2017). The probability operator used in our query language syntactically and semantically corresponds to the probability operator in Prob-$\mathcal{ALC}$ and Prob-$\mathcal{EL}$.

To our knowledge, the only work that combines both temporal and probabilistic query answering in the presence of description logic ontologies is (Ceylan and Peñaloza 2015). Albeit, the authors consider a different setting, in which the flow of time is modeled by a Markov-process, and not by a sequence of observations as in our case. Moreover, they do not consider a rich query language like ours, but focus on computing the probability that some axiom is entailed in some given time range. (Dylla, Miliaraki, and Theobald 2013) consider temporal probabilistic databases with temporal Datalog rules and constraints, and computing probabilities of conjunctive queries in these KBs. Both works do not allow for nested probabilities as part of the query language.

To handle scenarios like in our example, we propose a framework that combines the ideas from (Borgwardt and Thost 2015) for temporal knowledge bases with the framework for probabilistic knowledge bases introduced in (Jung and Lutz 2012). To query data in the resulting temporal probabilistic knowledge bases, our language extends temporal queries with probabilistic operators, to allow to assign probability bounds to arbitrary parts of the query. We establish a more or less complete picture of the complexity of query entailment in this framework for various DLs (see Figure 1, explained in detail throughout the text), and also discuss a restricted variant of our query language without negation, which sometimes leads to a restricted complexity.

Detailed proofs are provided in the extended version of the paper (Koopmann 2018).

Preliminaries

We recall the DLs studied in the paper, conjunctive query answering, and probabilistic complexity classes.

Description Logics. Let $\mathsf{N_C}$, $\mathsf{N_R}$ and $\mathsf{N_I}$ be pair-wise countably infinite sets of respectively *concept names*, *role names* and *individual names*. A *role* is an expression of the forms r, r^-, where $r \in \mathsf{N_R}$. *Concepts* are of the following forms, where $A \in \mathsf{N_C}$, R is a role, C, D are concepts, $n \in \mathbb{N}$ and $a \in \mathsf{N_I}$:

$$A \mid C \sqcap D \mid \exists R.C \mid \forall R.C \mid {\geq} nR.C \mid \{a\}.$$

A *TBox* is a set of *axioms* of the forms $C \sqsubseteq D$, $R \sqsubseteq S$ and $\mathsf{trans}(R)$, where C,D are concepts and R, S roles, while an ABox is a set of assertions of the forms $A(a)$ and $r(a, b)$, $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, $a, b \in \mathsf{N_I}$. For a TBox $\mathcal{T}$, we define the relation $\prec_{\mathcal{T}}$ s.t for two roles R, S, $S \prec_{\mathcal{T}} R$ holds if $S' \sqsubseteq R' \in \mathcal{T}$ with $S, S' \in \{s, s^-\}$ and $R, R' \in \{r, r^-\}$, $r, s \in \mathsf{N_R}$. A role R is *complex wrt.* $\mathcal{T}$ if $\mathsf{trans}(S) \in \mathcal{T}$ for some role S s.t. $S \prec^*_{\mathcal{T}} R$. To ensure decidability, we require for every concept of the form ${\geq}nR.C$ in $\mathcal{T}$ that R is not complex. Now a *knowledge base* (KB) is a tuple $\langle \mathcal{T}, \mathcal{A} \rangle$ of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. We differentiate different DLs based on the operators allowed: $\mathcal{EL}$ only supports concepts of the form A, $C \sqcap D$ and $\exists R.C$ and axioms of the form $C \sqsubseteq D$, no roles of the form r^-, and no axioms of the forms $R \sqsubseteq S$ or $\mathsf{trans}(R)$. $\mathcal{ALC}$ extends $\mathcal{EL}$ with concepts of the form $\neg C$, and $\mathcal{S}$ extends $\mathcal{ALC}$ with axioms of the form

$\mathsf{trans}(R)$. More expressive DLs are denoted by attaching a letter to the DL, where we use $\mathcal{I}$ for support of roles r^-, $\mathcal{O}$ for concepts of the form $\{a\}$, $\mathcal{Q}$ for concepts of the form $\geq nR.C$, and $\mathcal{H}$ for axioms of the form $R \sqsubseteq S$. For example, $\mathcal{SHI}$ extends $\mathcal{S}$ with axioms of the form $R \sqsubseteq S$ and roles of the form r^-, whereas $\mathcal{ALCHOQ}$ extends $\mathcal{ALC}$ with concepts of the form $\{a\}$ and $\geq nR.C$. Depending on the DL $\mathcal{L}$ used, we speak of $\mathcal{L}$ concepts, $\mathcal{L}$ axioms, $\mathcal{L}$ TBoxes and $\mathcal{L}$ KBs.

The semantics of KBs is defined in terms of *interpretations* $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, where $\Delta^{\mathcal{I}}$ is a set of *domain elements* and $\cdot^{\mathcal{I}}$ maps each concept name $A \in \mathsf{N_C}$ to a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, each role name $r \in \mathsf{N_R}$ to a relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, each individual name $a \in \mathsf{N_I}$ to a domain element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, and each role r^- to $(r^-)^{\mathcal{I}} = (r^{\mathcal{I}})^-$. It is extended to concepts as follows.

$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}, (\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}, \{a\}^{\mathcal{I}} = \{a^{\mathcal{I}}\},$$
$$(\exists R.C)^{\mathcal{I}} = \{d \in \mathcal{I} \mid \exists e \in \Delta^{\mathcal{I}} : \langle d, e \rangle \in R^{\mathcal{I}}, e \in C^{\mathcal{I}}\},$$
$$(\geq nR.C)^{\mathcal{I}} = \{d \in \mathcal{I} \mid \#\{e \in \Delta^{\mathcal{I}} \mid \langle d, e \rangle \in R^{\mathcal{I}}, e \in C^{\mathcal{I}}\} \geq n\}$$

We say that an interpretation $\mathcal{I}$ *satisfies* an axiom/assertion α, in symbols $\mathcal{I} \models \alpha$, if $\alpha = C \sqsubseteq D$ and $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$; $\alpha = R \sqsubseteq S$ and $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$; $\alpha = \mathsf{trans}(R)$ and $R^{\mathcal{I}} = (R^{\mathcal{I}})^+$; $\alpha = A(a)$ and $a^{\mathcal{I}} \in A^{\mathcal{I}}$; and $\alpha = r(a, b)$ and $\langle a^{\mathcal{I}}, b^{\mathcal{I}} \rangle \in r^{\mathcal{I}}$. $\mathcal{I}$ is a *model* of a TBox/ABox/KB iff it satisfies all axioms in it. Finally, a TBox/ABox/KB $\mathcal{X}$ *entails* an axiom/assertion α iff α is satisfied by every model of $\mathcal{X}$.

Conjunctive Queries. A *conjunctive query* (CQ) takes the form $q = \exists \vec{y}.\phi(\vec{x}, \vec{y})$, where $\vec{x}$, $\vec{y}$ are vectors of variables and $\phi(\vec{x}, \vec{y})$ is a conjunction over atoms of the forms $A(t_1)$ and $r(t_1, t_2)$, where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$ is not complex, and t_1 and t_2 are *terms* taken from $\mathsf{N_I}$, $\vec{x}$ and $\vec{y}$. $\vec{x}$ are the *answer variables* of q. Given an interpretation $\mathcal{I}$ and a CQ q with answer variables $x_1, \ldots, x_n$, the vector $a_1 \ldots a_n \subseteq {\mathsf{N_I}}^n$ is an *answer of q in $\mathcal{I}$* if there exists a mapping $\pi : \mathsf{term}(q) \to \Delta^{\mathcal{I}}$ s.t. $\pi(x_i) = a_i$ for $i \in [\![1, n]\!]$, $\pi(b) = b^{\mathcal{I}}$ for $b \in \mathsf{N_I}$, $\pi(t) \in A^{\mathcal{I}}$ for every $A(t)$ in q, and $\langle \pi(t_1), \pi(t_2) \rangle \in r^{\mathcal{I}}$ for every $r(t_1, t_2)$ in q. A vector $a_1 \ldots a_n$ is a *certain answer* of q in a KB $\mathcal{K}$ if it is an answer in every model of $\mathcal{K}$. If a query does not contain any answer variables, it is a *Boolean CQ*, and we say it is *entailed* by a KB $\mathcal{K}$ (interpretation $\mathcal{I}$) if it has the empty vector as answer. The complexity of query entailment for KBs in various DLs is shown in the left-part of Figure 1.

Probabilistic Complexity Classes. The complexity class PP is defined using *probabilistic Turing machines*, which are like non-deterministic Turing machines, but with an alternative acceptance condition: namely, they accept an input iff at least half of the computation paths end in an accepting state. PP describes the class of all problems that can be decided by a probabilistic Turing machine in which all paths are polynomially bounded by the size of the input. By using oracles, we can obtain the complexity classes $\mathrm{PP}^{\mathrm{NP}}$ and P^{PP}, for which we have the relations $\mathrm{NP} \cup \mathrm{coNP} \subseteq \mathrm{PP} \subseteq \mathrm{PP}^{\mathrm{NP}} \subseteq \mathrm{P}^{\mathrm{PP}} \subseteq \mathrm{PSPACE}$ (Toda 1991). Strongly related to the decision class PP is the function class #P, which is the class of functions that can be computed by counting accepting paths in a non-deterministic polynomial time-bounded Turing machine.

Temporal Probabilistic Knowledge Bases and Queries

We introduce our framework for temporal probabilistic query answering.

Temporal Probabilistic Knowledge Bases. Regarding the probabilistic aspects, we follow the paradigm introduced in (Jung and Lutz 2012) for atemporal probabilistic KBs. To keep things simple, we focus on the easiest type of probabilistic fact bases presented there, the so-called *assertion-independent probabilistic ABoxes* (ipABoxes). Here, assertions are assigned probabilities, which are assumed to be statistically independent. They correspond to the tuple-independent probabilistic databases studied in (Dalvi and Suciu 2007).

An *ipABox* is a set of *probabilistic ABox assertions* of the form α: p, where α is an ABox assertion and p is a probability value between 0 and 1. Intuitively, α: p expresses that the assertion α holds with a probability of at least p. Instead of α: 1, we may just write α if the meaning is clear from the context. ipABoxes only specify a *lower bound* on the probability, to conform with the open-world semantics common in ontology-based representations. This means, we might infer using other information in the KB that the probability is in fact higher.[1] A temporal probabilistic KB (TPKB) is now a tuple $\langle \mathcal{T}, (\mathcal{A}_i)_{i \in [\![1,n]\!]} \rangle$, where $\mathcal{T}$ is a TBox and $(\mathcal{A}_i)_{i \in [\![1,n]\!]}$ is a sequence of ipABoxes.

We define the semantics of TPKBs using the *possible worlds semantics*, as common to probabilistic logics and databases (Dalvi and Suciu 2007). For a given a TPKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{i \in [\![1,n]\!]} \rangle$, the set $\Omega_{\mathcal{K}}$ of *possible worlds of* $\mathcal{K}$ contains all sequences $w = (\mathcal{A}'_i)_{i \in [\![1,n]\!]}$ of classical ABoxes such that for every $i \in [\![1, n]\!]$ and $\alpha \in \mathcal{A}'_i$, $\mathcal{A}_i$ contains an axiom of the form α: p. Each TPKB uniquely defines a probability space $\langle \Omega_{\mathcal{K}}, \mu_{\mathcal{K}} \rangle$, where the probability measure $\mu_{\mathcal{K}} : 2^{\Omega_{\mathcal{K}}} \to [0, 1]$ satisfies

$$\mu_{\mathcal{K}}(\{(\mathcal{A}'_i)_{i \in [\![1,n]\!]}\}) = \prod_{\substack{i \in [\![1,n]\!] \\ \alpha:\, p \in \mathcal{A}_i \\ \alpha \in \mathcal{A}'_i}} p \cdot \prod_{\substack{i \in [\![1,n]\!] \\ \alpha:\, p \in \mathcal{A}_i \\ \alpha \notin \mathcal{A}'_i}} (1 - p)$$

and for $W \subseteq \Omega_{\mathcal{K}}$, $\mu_{\mathcal{K}}(W) = \sum_{w \in W} \mu(\{w\})$. Intuitively, $\mu_{\mathcal{K}}(W)$ gives the probability of being in one of the

[1] This is different to the open-world semantics for probabilistic databases suggested in (Ceylan, Darwiche, and van den Broeck 2017), which assumes a fixed upper probability for facts absent in the data.

$\Omega_\mathcal{K}$	$\mathcal{A}'_1$	$\mathcal{A}'_3$	$\mathcal{A}'_4$	$\mu_\mathcal{K}$
w_1	$\mathsf{hBP}(p,b), \mathsf{HBP}(b)$	$\mathsf{HBP}(b)$	$\mathsf{HBP}(b)$	0.378
w_2	$\mathsf{hBP}(p,b)$	$\mathsf{HBP}(b)$	$\mathsf{HBP}(b)$	0.162
w_3	$\mathsf{hBP}(p,b), \mathsf{HBP}(b)$	$\emptyset$	$\mathsf{HBP}(b)$	0.042
w_4	$\mathsf{hBP}(p,b)$	$\emptyset$	$\mathsf{HBP}(b)$	0.018
w_5	$\mathsf{hBP}(p,b), \mathsf{HBP}(b)$	$\mathsf{HBP}(b)$	$\emptyset$	0.252
w_6	$\mathsf{hBP}(p,b)$	$\mathsf{HBP}(b)$	$\emptyset$	0.108
w_7	$\mathsf{hBP}(p,b), \mathsf{HBP}(b)$	$\emptyset$	$\emptyset$	0.028
w_8	$\mathsf{hBP}(p,b)$	$\emptyset$	$\emptyset$	0.012

Table 1: Probability space of example TPKB.

possible worlds in W, by summing up the probabilities of each possible world. The definition of $\mu_\mathcal{K}(W)$ reflects the assumption that all probabilities in the TPKB are statistically independent.

Example 1. We define the TPKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{i \in [\![1,5]\!]} \rangle$, where $\mathcal{T}$ contains the axiom

$$\mathsf{HighBloodPressurePatient} \equiv \exists \mathsf{hasBloodPressure}.\mathsf{HighBloodPressure},$$

and the probabilistic ABoxes are

$$\mathcal{A}_1 = \{\mathsf{hasBloodPressure}(a,b), \mathsf{HighBloodPressure}(b){:}0.7\}$$
$$\mathcal{A}_3 = \{\mathsf{HighBloodPressure}(b){:}\ 0.9\}$$
$$\mathcal{A}_4 = \{\mathsf{HighBloodPressure}(b){:}\ 0.6\}$$

and $\mathcal{A}_2 = \mathcal{A}_5 = \emptyset$. Every possible world $w = (\mathcal{A}'_i)_{i \in [\![1,5]\!]}$ with $\mathsf{hasBloodPressure}(a,b) \notin \mathcal{A}'_1$ has probability $\mu_\mathcal{K}(w) = 0$. The remaining possible worlds, excluding time points 2 and 5, are shown in Figure 1, with the probability measure $\mu_\mathcal{K}$ shown in the last column, where hBP is short for $\mathsf{hasBloodPressure}$ and HBP is short for $\mathsf{HighBloodPressure}$. □

Remark 1. We follow the semantical idea of ipABoxes, in which all assertions are assumed to be statistically independent, mainly to keep the representation simple. Of course, for realistic applications, the assumption that all probabilistic facts are statistically independent is not always accurate, and already (Jung and Lutz 2012) specify a more general concept of probabilistic ABoxes. The complexity upper bounds established in this paper only rely on a fixed probability distribution over the possible worlds, which is why they can be easily extended to more refined settings. □

Based on the probability measure, we define models by assigning to each possible world a sequence of classical interpretations. A *model* of a TPKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{i \in [\![1,n]\!]} \rangle$ is a mapping ι from possible worlds $w = (\mathcal{A}'_i)_{i \in [\![1,n]\!]} \in \Omega_\mathcal{K}$ to sequences $(\iota(w)_i)_{i>0}$ of (classical) models of $\mathcal{T}$ s.t. for all $i \in [\![1,n]\!]$, $\iota(w)_i$ is a model of the classical knowledge base $\langle \mathcal{T}, \mathcal{A}'_i \rangle$, and all $\iota(w)_i$ have the same set Δ^ι of domain elements (constant domain assumption).

Rigid Names. As typical for temporal knowledge bases, we may assume in addition a set N_rig of *rigid names*, containing the set $\mathsf{N}_\mathsf{Crig} \subseteq \mathsf{N}_\mathsf{C}$ of *rigid concept names* and the set $\mathsf{N}_\mathsf{Rrig} \subseteq \mathsf{N}_\mathsf{R}$ of *rigid role names*. Rigid names denote names whose interpretation is independent of the flow of time. We say that a model ι of a TPKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{i \in [\![1,n]\!]} \rangle$ *respects rigid names* iff for all $w \in \Omega_\mathcal{K}$, $i, j \in [\![1,n]\!]$ and $X \in \mathsf{N}_\mathsf{rig}$, $X^{\iota(w)_i} = X^{\iota(w)_j}$. Allowing for rigid names often has a direct impact on complexity and decidability of common reasoning problems, which is why typically different cases based on whether $\mathsf{N}_\mathsf{Crig} = \emptyset$ or $\mathsf{N}_\mathsf{Rrig} = \emptyset$ are studied for complexity.

Example 2. In the above example, the relation hasBP is rigid, as its interpretation should be independent of time, while the concept HighBP is not rigid, as the blood pressure of a patient can change from high to not high. As a consequence, the individual p will be related to the blood pressure b at all time points, even though the assertion $\mathsf{hasBP}(p,b)$ only occurs in $\mathcal{A}_1$. □

Temporal Probabilistic Queries. To query temporal data in the OBDA framework, extensions of conjunctive queries with operators from linear temporal logic (LTL) have been considered and well-investigated as *temporal queries* (TQs) (Borgwardt and Thost 2015). When applied on a temporal probabilistic KB, an assignment of the answer variables in a TQ becomes an answer with a certain probability, depending on the query, we might be interested only in answers that holds with a certain minimal or maximal probability. Rather than just assigning an overall probability threshold, we might want to mark parts of the query with different probability upper and lower bounds. For example, in the scenario sketched in the introduction, the smart-phone could be equipped with a motion sensor to detect the probability that the patient is currently exercising, and one might want to detect situations in which the probability of them exercising is low, while the probability of his blood pressure being above some threshold is high.

To be able to describe all this, we extend TQs with probability operators. A *temporal probabilistic query* (TPQ) is of one of the following forms, where q is a CQ, ϕ_1 and ϕ_2 are a TPQs, $p \in [0,1]$ and $\sim \in \{<, \leq, =, \geq, >\}$.

$$q \mid \neg\phi_1 \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \bigcirc\phi_1 \mid \Diamond\phi_1 \mid \Box\phi_1 \mid \phi_1 \mathcal{U} \phi_2$$
$$\bigcirc^-\phi_1 \mid \Diamond^-\phi_1 \mid \Box^-\phi_1 \mid \phi_1 \mathcal{S} \phi_2 \mid \mathsf{P}_{\sim p}\phi_1$$

The propositional operators, $\bigcirc$ (next), $\Diamond$ (eventually), $\mathcal{U}$ (until) are found in TQs, while $\mathsf{P}_{\sim p}$ is the operator that we add to this language. Note that due the disjunction operator, we can also express *unions of conjunctive queries* (UCQs), which are simply disjunctions of CQs. The answer variables of a TPQ ϕ are the answer variables of the CQs in ϕ. A TPQ ϕ is *Boolean* if every variable in ϕ is bound by an existential quantifier.

In order to define the semantics of TPQs, we have to take into consideration the two dimensions in which queries refer to a temporal probabilistic model. While the temporal operators refer to the time line in a single possible world, for the probability operators we have

ϕ	$\iota, w, i \models \phi$ iff
$\exists\vec{y}.\psi(\vec{y})$	$\iota(w), i \models \exists\vec{y}.\psi(\vec{y})$
$\neg\phi_1$	$\iota, w, i \not\models \phi_1$
$\phi_1 \wedge \phi_2$	$\iota, w, i \models \phi_1$ and $\iota, w, i \models \phi_2$
$\phi_1 \vee \phi_2$	$\iota, w, i \models \phi_1$ or $\iota, w, i \models \phi_2$
$\bigcirc\phi_1$	$\iota, w, i+1 \models \phi_1$
$\bigcirc^-\phi_1$	$\iota, w, i-1 \models \phi_1$ and $i > 0$
$\Diamond\phi_1$	$\iota, w, j \models \phi_1$ for some $j \geq i$
$\Diamond^-\phi_1$	$\iota, w, j \models \phi_1$ for some $j \leq i$
$\Box\phi_1$	$\iota, w, j \models \phi_1$ for all $j \geq i$
$\Box^-\phi_1$	$\iota, w, j \models \phi_1$ for all $j \leq i$
$\phi_1\,\mathcal{U}\phi_2$	$\iota, w, j \models \phi_2$ for some $j \geq i$, and $\iota, w, k \models \phi_1$ for all $k \in \llbracket i, j-1 \rrbracket$
$\phi_1\mathcal{S}\phi_2$	$\iota, w, j \models \phi_2$ for some $j \leq i$, and $\iota, w, k \models \phi_1$ for all $k \in \llbracket j+1, i \rrbracket$
$\mathsf{P}_{\sim p}\phi$	$\mu_{\mathcal{K}}(\{w' \in \Omega_{\mathcal{K}} \mid \iota, w', i \models \phi\}) \sim p$, where $\sim \in \{<, \leq, =, \geq, >\}$

Table 2: Entailment of Boolean TPQs under interpretation ι.

to aggregate the probabilities of the possible worlds in which a query is entailed.

Let $\mathcal{K}$ be a TPKB, ι a model of $\mathcal{K}$, and ϕ a Boolean TPQ. For a single possible world $w \in \Omega_{\mathcal{K}}$ and a time point i, we say that ϕ is satisfied at w, i under ι, in symbols $\iota, w, i \models \phi$ iff the conditions in Table 2 are satisfied. Note that the temporal operators refer to the time line of a single possible world, for which they are defined as in (Borgwardt and Thost 2015). In contrast, the probability operators refer to the current time point in multiple possible worlds, and are defined similar to the probabilistic concept constructor in Prob-$\mathcal{ALC}$ (Gutiérrez-Basulto et al. 2017). A Boolean TPQ ϕ is *satisfied* in an interpretation ι at i, in symbols $\iota, i \models \phi$, iff $\iota, w, i \models \phi$ for all $w \in \Omega_{\mathcal{K}}$. It is *entailed* by the TPKB $\mathcal{K}$ at i, in symbols $\mathcal{K}, i \models \phi$, iff $\iota, i \models \phi$ for all models ι of $\mathcal{K}$. ϕ is *satisfiable in $\mathcal{K}$ at i* iff there exists a model ι of $\mathcal{K}$ s.t. $\iota, i \models \phi$. Note that satisfiability is complementary to entailment: namely, ϕ is satisfiable in $\mathcal{K}$ at i iff $\mathcal{K}, i \not\models \neg\phi$.

Now given a TPKB $\mathcal{K}$, a TPQ ϕ with answer variables $\vec{x}$, a time point $i > 0$, and a mapping $\sigma : \vec{x} \to \mathsf{N_I}$, σ is a *certain answer for ϕ in $\mathcal{K}$ at i* iff $\mathcal{K}, i \models \phi'$, where ϕ' is the result of applying σ on ϕ. As common, since computing answers for TPQs can be seen as a search problem that uses Boolean TPQ entailment, we focus on the decision problem of query entailment, and may refer to Boolean TPQs simply as TPQs.

Example 3. We consider a slight variation of the query from the introduction.

$$\mathsf{P}_{>.8}(\bigcirc^{-5}\Diamond(\mathsf{HighBPP}(x) \wedge \Diamond\mathsf{HighBPP}(x)))$$

For $x = p$ and time point 5, the query below the probability operator is entailed in every model of the possible worlds w_1, w_2, w_3 and w_5, which together have a probability of 0.834. Consequently, p is an answer to the query at time point 5. Now consider the variation where the probability operators are moved inside.

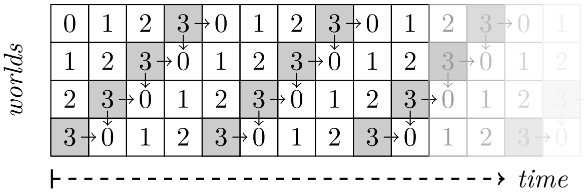

Figure 2: Illustration of how the counters are used in the possible worlds.

$$\bigcirc^{-5}\Diamond(\mathsf{P}_{>.8}(\mathsf{HighBPP}(x)) \wedge \Diamond\mathsf{P}_{>.8}(\mathsf{HighBPP}(x)))$$

This corresponds to the situation where at least twice in the last 5 time units, the probability of having a high blood pressure was above 0.8. As this probability is only once above this bound, this query is not entailed.

The complexity of TPQ entailment for various DLs is shown in Figure 1, where we compare against the complexity of classical query entailment (left column), and distinguish the cases based on whether $\mathsf{N_{Rrig}} = \emptyset$ (middle column) or $\mathsf{N_{Rrig}} \neq \emptyset$ (right column). All complexities remain tight independent on whether we admit rigid concept names ($\mathsf{N_{Crig}} \neq \emptyset$). Note that the ExpSpace-result for $\mathcal{ELH}$ remains tight even without any TBox. Results marked with (pos) regard *positive TPQs*, which we discuss towards the end of the paper.

Hardness of TPQ Entailment

We show that TPQ satisfiability, and thus entailment, is ExpSpace-hard even if $\mathcal{T} = \emptyset$ and $\mathsf{N_{Crig}} = \mathsf{N_{Rrig}} = \emptyset$, by reduction of the exponential variant of the corridor tiling problem (van Emde Boas 1997). In this problem, we are given a set T of *tile types*, two special tile types t_s,$t_e \in T$, a natural number n, and two functions v and h of *compatibility constraints* $v : T \to 2^T$ (vertical) and $h : T \to 2^T$ (horizontal). The input is an *instance* of the exponential corridor tiling problem if there exists a number $m \in \mathbb{N}$ and a *tiling* $f : \llbracket 0, m \rrbracket \times \llbracket 0, 2^n - 1 \rrbracket \to T$ such that $f(0,0) = t_s$, $f(m,0) = t_e$, and for all $x \in \llbracket 0, m \rrbracket$ and $y \in \llbracket 0, 2^n - 1 \rrbracket$, if $x < m$, $f(x+1,y) \in h(f(x,y))$ and if $y < 2^n - 1$, $f(x,y+1) \in v(f(x,y))$.

We use n concept names A_i to mark the different possible worlds $w \in \Omega_{\mathcal{K}}$ with a counter, such that in interpretations ι that satisfy both the TPQ and the TPKB, $\iota, w, j \models A_i(a)$ iff the ith bit of the counter is 1 at time point j, and $\iota, w, j \not\models A_i(a)$ iff the ith bit is 0 at time point j. The ipABox $\mathcal{A}_1 = \{A_i(a) \sim 0.5 \mid i \in \llbracket 1, n \rrbracket\}$ assigns every possible world a different counter value. Our query makes sure that the counter values are increased for each time point. Figure 2 illustrates this idea. Each possible world corresponds to a row in the tiling, with its counter value at time point 1 denoting the row number.

At each time point, two possible worlds can be recognised by simple queries: the one whose counter value is

0 (which satisfies $\bigwedge_{1\leq i\leq n}\neg A_i(a)$), and the one whose counter value is 2^n-1 (which satisfies $\bigwedge_{1\leq i\leq n} A_i(a)$). Unless the latter one represents the last row, these worlds correspond to neighbours in the tiling, which means that for these worlds, we can enforce the vertical tiling conditions with the following query, where $L(a)$ is an assertion that marks the last row, and for a tile type $t\in T$, $B_t(a)$ expresses that the current cell has a tile of type t.

$$\Box\bigwedge_{t_1\in T}\left(\left(B_{t_1}(a)\wedge\bigwedge_{i\in[\![1,n]\!]}A_i(a)\wedge\neg L(a)\right)\right.$$
$$\left.\to\bigvee_{t_2\in v(t_1)}\mathsf{P}_{=1}\left(\left(\bigwedge_{i\in[\![1,n]\!]}\neg A_i(a)\right)\to B_{t_2}(a)\right)\right)$$

As we can only check the vertical tiling conditions for one pair of rows at a time, we represent each cell by up to 2^n succeeding time points in each possible world, switching to the next tile only when the counter reaches 2^n-1. The remaining details of the reduction can be found in the extended version of the paper. The hardness for $\mathcal{ALC}$ with rigid roles follows from the non-probabilistic case (Baader, Borgwardt, and Lippmann 2015).

Theorem 1. *The lower bounds regarding general TPQs in Figure 1 hold.*

Deciding TPQ Entailment

We show the complexity upper bounds for general queries shown in Figure 1, where we again focus on the complementary problem of query satisfiability.

The main idea is to define appropriate abstractions of models of the TPKB $\mathcal{K}=\langle\mathcal{T},(\mathcal{A}_i)_{i\in[\![1,n]\!]}\rangle$ which we call *quasi-models*, and then show how an abstraction that witnesses the satisfiability can be guessed and verified within the targeted complexity bound. We first define the structure to represent single time points, which we call *quasi-states.* We can assume without loss of generality that ϕ contains only the operators $\wedge$, $\neg$, $\mathcal{U}$, $\mathcal{S}$ and $\mathsf{P}_{\sim p}$, since the remaining operators can be linearly encoded using known equivalences. Denote by $\mathsf{sub}(\phi)$ the subqueries of ϕ and set $T(\phi)=\{\psi,\neg\psi\mid\psi\in\mathsf{sub}(\phi)\}$. A quasi-state is now a mapping $Q:\Omega_{\mathcal{K}}\to T(\phi)$ that satisfies the following conditions:

S1 $\neg\psi\in Q(w)$ iff $\psi\notin Q(w)$,

S2 for all $\psi_1\wedge\psi_2\in T(\phi)$: $\psi_1\wedge\psi_2\in Q(w)$ iff $\psi_1\in Q(w)$ and $\psi_2\in Q(w)$, and

S3 for all $\mathsf{P}_{\sim p}(\psi)\in T(\phi)$: $\mathsf{P}_{\sim p}(\psi)\in Q(w)$ iff $\mu_{\mathcal{K}}(\{w\mid\psi\in Q(w)\})\sim p$.

The quasi-state abstracts probabilistic interpretations at a single time point by assigning queries to each possible world according to the semantics of the atemporal operators in our query language. To incorporate the temporal dimension, we consider unbounded sequences of quasi-states $(Q_i)_{i\geq 1}$, which we call *quasi-models for* $\mathcal{K}$, and which have to satisfy the following conditions for $i\geq 1$ and $w=(\mathcal{A}'_i)_{i\in[\![1,n]\!]}\in\Omega_{\mathcal{K}}$:

Q1 if $i\in[\![1,n]\!]$, then $\langle\mathcal{T},\mathcal{A}'_i\rangle\not\models\neg\left(\bigwedge_{\psi\in X}\psi\right)$, where $X=\{\psi\in Q_i(w)\mid\psi$ is a CQ or a negated CQ$\}$.

Q2 for all $\bigcirc\psi\in T(\phi)$, $\bigcirc\psi\in Q_i(w)$ iff $\psi\in Q_{i+1}(w)$,

Q3 for all $\bigcirc^{-}\psi\in T(\phi)$, $\bigcirc^{-}\psi\in Q_{i+1}(w)$ iff $\psi\in Q_i(w)$,

Q4 for all $\psi_1\mathcal{U}\psi_2\in T(\phi)$, $\psi_1\mathcal{U}\psi_2\in Q_i$ iff there exists $j\geq i$ s.t. $\psi_2\in Q_j(w)$ and for all $k\in[\![i,j-1]\!]$, $\psi_1\in Q_k(w)$, and

Q5 for all $\psi_1\mathcal{S}\psi_2\in T(\phi)$, $\psi_1\mathcal{S}\psi_2\in Q_i$ iff there exists $j\leq i$ s.t. $\psi_2\in Q_j(w)$ and for all $k\in[\![j-1,i]\!]$, $\psi_1\in Q_k(w)$.

Again, the intuition behind these conditions is given directly by the semantics of the temporal operators.

To handle rigid names, we need an additional structure to make sure that the queries assigned to different time points in a possible world correspond to a sequence of interpretations that respects rigid names. Let $\{q_1,\dots,q_n\}$ be the CQs that occur in the query ϕ. For each $w\in\Omega_{\mathcal{K}}$, we guess the set $\mathcal{S}(w)\subseteq 2^{\{q_1,\dots,q_m\}}$ of *sets of queries* that are allowed be satisfied together at a time point in w, and thus obtain a mapping $\mathcal{S}:\Omega_{\mathcal{K}}\to 2^{2^{\{q_1,\dots,q_m\}}}$. To be consistent with the rigid names, $\mathcal{S}(w)$ has to correspond to a set of interpretations that agree on the rigid names, where each set of queries corresponds to one interpretation. To also take into account the ABoxes, we use a second mapping $a:\Omega_{\mathcal{K}}\times[\![1,n]\!]\to 2^{\{q_1,\dots,q_n\}}$, which for each $w\in\Omega_{\mathcal{K}}$ assigns elements from $\mathcal{S}(w)$ to the ABoxes in w. Given such mappings $\mathcal{S}$ and a, we say that a quasi-model $(Q_i)_{i\geq 1}$ is *compatible to* $\mathcal{S}$ *and* a if for every $i\geq 0$ and $w\in\Omega_{\mathcal{K}}$:

Q6 $Q_i(w)\cap\{q_1,\dots,q_m\}\in\mathcal{S}(w)$, and

Q7 if $i\in[\![1,n]\!]$, $Q_i(w)\cap\{q_1,\dots,q_m\}=a(w,i)$.

The following definition captures when $\mathcal{S}$ and a correspond to a model of $\mathcal{K}$ that respects rigid names.

Definition 4. Let $w=(\mathcal{A}'_i)_{i\in[\![1,n]\!]}\in\Omega_{\mathcal{K}}$, $\mathcal{S}:\Omega_{\mathcal{K}}\to 2^{2^{\{q_1,\dots,q_m\}}}$ and $a:\Omega_{\mathcal{K}}\times[\![1,n]\!]\to 2^{\{q_1,\dots,q_m\}}$, where $\mathcal{S}(w)=\{X_1,\dots,X_k\}$. Then, $\mathcal{S}$ is called *r-satisfiable wrt.* w *and* a iff there exist (classical) interpretations $\mathcal{J}_1,\dots,\mathcal{J}_k,\mathcal{I}_1,\dots,\mathcal{I}_n$ such that

R1 the interpretations are models of $\mathcal{T}$,

R2 for any two interpretations $\mathcal{I},\mathcal{I}'\in\{\mathcal{J}_1,\dots,\mathcal{J}_k,\mathcal{I}_1,\dots,\mathcal{I}_n\}$, we have $\Delta^{\mathcal{I}}=\Delta^{\mathcal{I}'}$ and $X^{\mathcal{I}}=X^{\mathcal{I}'}$ for all $X\in\mathsf{N}_{\mathsf{rig}}$,

R3 for all $i\in[\![1,k]\!]$, $\mathcal{J}_i\models\bigwedge_{q\in X_i}q\wedge\bigwedge_{q\notin X_i}\neg q$, and

R4 for all $i\in[\![1,n]\!]$, $\mathcal{I}_i\models\bigwedge_{q\in a(w,i)}q\wedge\bigwedge_{q\notin a(w,i)}\neg q$ and $\mathcal{I}_i\models\mathcal{A}'_i$.

$\mathcal{S}$ is r-satisfiable wrt. a, if for all $w\in\Omega_{\mathcal{K}}$, $\mathcal{S}$ is r-satisfiable wrt. w and a.

Note that the interpretations $\mathcal{J}_1,\dots,\mathcal{J}_k$ in the interpretation correspond to the elements $\{X_1,\dots,X_k\}=\mathcal{S}(w)$, so that Condition **R2** ensures that we can find sequences of interpretation that respect rigid names.

Lemma 5. *Wrt. the size of* $\mathcal{K}$ *and* ϕ*, r-satisfiability for* $\mathcal{L}$*-TPKBs can be decided in*

1. NExpTime *for* $\mathcal{L} = \mathcal{ELH}$,
2. NExpTime *for* $\mathcal{L} = \mathcal{SHQ}$ *if* $\mathsf{N_{Rrig}} = \emptyset$,
3. 2-ExpTime *for* $\mathcal{L} \in \{\mathcal{SHIQ}, \mathcal{SHOQ}, \mathcal{SHOI}\}$, *and*
4. *it is decidable for* $\mathcal{L} = \mathcal{ALCHOIQ}$.

Proof(Sketch). We define a classical KB based on the mappings a and a world $w \in \Omega_\mathcal{K}$ which encodes the interpretation of non-rigid names $Y \in (\mathsf{N_C} \cup \mathsf{N_R}) \setminus \mathsf{N_{rig}}$ for different elements $X_i \in \mathcal{S}_i(w)$ using fresh names Y^i. A similar translation is applied to the CQs $q \in X_i$. We can then reduce the properties in Definition 4 to a query entailment problem, where the KB and the query are of exponential size with respect to the input. While query entailment for $\mathcal{ALCI}$ and $\mathcal{ALCO}$ is 2-ExpTime-hard (Lutz 2007; Ngo, Ortiz, and Simkus 2016), we obtain by inspection of the procedures in (Glimm et al. 2008; Glimm, Horrocks, and Sattler 2008; Calvanese, Eiter, and Ortiz 2009) that this particular query entailment test can be performed in 2-ExpTime. For $\mathcal{SHQ}$ and $\mathsf{N_{Rrig}} = \emptyset$, the complexity follows from results in (Baader, Borgwardt, and Lippmann 2015). □

Quasi-models are indeed sufficient to witness the satisfiability of a TPQ. If the quasi-model is additionally compatible to mappings $\mathcal{S}$ and a s.t. $\mathcal{S}$ is r-satisfiable wrt. a, then they witness the satisfiability of a TPQ from TKBs with rigid names. Crucially for our complexity result, it is further sufficient to focus on quasi-models that have a regular shape.

Lemma 6. *ϕ is satisfiable in $\mathcal{K}$ at time point i iff there exists mappings $\mathcal{S} : \Omega_\mathcal{K} \to 2^{2^{\{q_1,\dots,q_m\}}}$ and $a : \Omega_\mathcal{K} \times [\![1, n]\!] \to 2^{\{q_1,\dots,q_m\}}$ and a quasi-model $(Q_j)_{j\geq 1}$ for $\mathcal{K}$ such that*

1. *$\phi \in Q_i(w)$ for all $w \in \Omega_\mathcal{K}$,*
2. *$(Q_j)_{j\geq 1}$ is compatible with $\mathcal{S}$ and a,*
3. *$\mathcal{S}$ is r-satisfiable wrt. to a, and*
4. *$(Q_j)_{j\geq 1}$ is of the form $Q_1, \dots Q_m(Q_{m+1}, \dots Q_{m+o})^\omega$, where m and o are both double exponentially bounded in the size of $\mathcal{K}$.*

Theorem 2. *The complexity upper bounds for general TPQs in Figure 1 hold.*

Proof (Sketch). We first guess the numbers m and o from Lemma 6. If $\mathsf{N_{rig}} \neq \emptyset$, we additionally guess the mappings $\mathcal{S}$ and a and verify that $\mathcal{S}$ is r-satisfiable wrt. a. We now guess the quasi-states Q_1, …, Q_{m+o} one after the other, where we carefully make sure that all the conditions in the definition of quasi-states and quasi-models are satisfied, and verify that Q_{m+o} is compatible to Q_{m+1}. This procedure runs in exponential space if r-satisfiability can be decided in exponential space, and in double exponential time if deciding r-satisfiability requires double exponential time. □

Positive TPQs

It turns out that for $\mathcal{EL}$, we can obtain better complexity bounds if we restrict ourselves to *positive TPQs*, which are TPQs that do not use the operators $\neg$, $\mathsf{P}_{<p}$ and $\mathsf{P}_{=p}$. The probability operators $\mathsf{P}_{<p}$ and $\mathsf{P}_{=p}$ can be seen as implicit negation operators, as they express the non-entailment of a query ϕ in some possible worlds, whereas $\mathsf{P}_{>p}\phi$ only expresses the positive entailment of ϕ in some possible worlds. The examples used in this paper all use only positive TPQs.

Definition 7. A TPQ is *positive* iff it does not use the operators $\neg$, $\mathsf{P}_{<p}$, $\mathsf{P}_{\leq p}$ and $\mathsf{P}_{=p}$.

For DLs that have negation, our reduction used to show ExpSpace-hardness can be adapted to query entailment for positive TPQs. As we reduced the corridor tiling problem to query satisfiability, the corresponding query entailment problem is of the form $\mathcal{K} \models \neg\phi$, where ϕ is the defined query. By pushing negations inside, we obtain a query in which every probability operator is of the form $\mathsf{P}_{>0}$ or $\mathsf{P}_{\geq 1}$, and negation only occurs in front of concept names. Therefore, for any DL extending $\mathcal{ALC}$, the complexity bounds established in the last sections remain tight even for positive TPQs. In contrast to $\mathcal{ALC}$, $\mathcal{EL}$ has the canonical model property, which makes it possible to test for entailment in different possible worlds independently. This allows for a strategy in which the TPQ is evaluated "inside out", by first evaluating the most nested probability operators, and then proceeding on the next level. Due to the known closure properties of the complexity class PP, we obtain a $\mathrm{PP}^{\mathrm{NP}}$ complexity upper bound if the nesting depth of the probability operators is bound, which we show to be tight, and otherwise a P^{NP} upper bound. This approach further allows us to establish tight complexity for *data complexity*, where the size of the query is assumed to be fix, marked in Figure 1 with (pos,dat).

Theorem 3. *The complexity results regarding positive TPQs in Figure 1 hold.*

Conclusion

We introduced a framework for representing and querying temporal probabilistic data within the ontology-based query answering paradigm, and established tight complexity bounds for most common description logics. While for expressive DLs starting from $\mathcal{ALCI}$ and $\mathcal{ALCO}$, adding both the temporal and the probabilistic dimension comes at no additional cost compared to classical query answering, for $\mathcal{ALC}$ and below, reasoning becomes harder both in comparison to purely temporal and purely probabilistic query answering. For instance, probabilistic query answering is ExpTime-complete for $\mathcal{ALC}$ and $\mathrm{PP}^{\mathrm{NP}}$-complete for $\mathcal{EL}$, and for $\mathsf{N_{Rrig}} = \emptyset$, it is ExpTime-complete for $\mathcal{ALC}$ and PSpace-complete for $\mathcal{EL}$, which contrasts with our ExpSpace-hardness that occurs already without a TBox. For $\mathcal{EL}$, this situation can be improved if we forbid negation in the query language, in which case temporal probabilistic query answering is not harder as in the atemporal case. We believe that our technique for showing the upper bound here could also be used for practical implementations. We are currently looking at how query rewriting techniques for simpler DLs such as *DL-Lite* could be used for this in connection with existing probabilistic database systems.

Acknowledgements

This work is supported by the German Research Foundation (DFG) within the Collaborative Research Center SFB 912 HAEC.

References

Artale, A.; Kontchakov, R.; Kovtunova, A.; Ryzhikov, V.; Wolter, F.; and Zakharyaschev, M. 2017. Ontology-mediated query answering over temporal data: A survey (invited talk). In *Proc. TIME 2017*, 1:1–1:37.

Baader, F.; Borgwardt, S.; Koopmann, P.; Ozaki, A.; and Thost, V. 2017. Metric temporal description logics with interval-rigid names. In *Proc. FroCoS 2017*, 60–76. Springer International.

Baader, F.; Borgwardt, S.; and Lippmann, M. 2015. Temporal query entailment in the description logic $\mathcal{SHQ}$. *J. Web Sem.* 33:71–93.

Baader, F.; Ghilardi, S.; and Lutz, C. 2012. LTL over description logic axioms. *ACM Trans. Comput. Log.* 13(3):21:1–21:32.

Baader, F.; Koopmann, P.; and Turhan, A. 2017. Using ontologies to query probabilistic numerical data. In *Proc. FroCoS 2017*, 77–94. Springer International.

Borgwardt, S., and Thost, V. 2015. Temporal query answering in the description logic $\mathcal{EL}$. In *Proc. IJCAI 2015*, 2819–2825. AAAI Press.

Borgwardt, S.; Ceylan, İ. İ.; and Lukasiewicz, T. 2017. Ontology-mediated queries for probabilistic databases. In *Proc. AAAI 2017*, 1063–1069. AAAI Press.

Brandt, S.; Kalayci, E. G.; Kontchakov, R.; Ryzhikov, V.; Xiao, G.; and Zakharyaschev, M. 2017. Ontology-based data access with a Horn fragment of metric temporal logic. In *Proc. AAAI 2017*, 1070–1076. AAAI Press.

Calvanese, D.; Eiter, T.; and Ortiz, M. 2009. Regular path queries in expressive description logics with nominals. In *Proc. IJCAI 2009*, 714–720. AAAI Press.

Ceylan, İ. İ., and Peñaloza, R. 2015. Dynamic Bayesian description logics. In *Proc. DL 2015*. CEUR-WS.

Ceylan, İ. İ., and Peñaloza, R. 2017. The Bayesian ontology language $\mathcal{BEL}$. *J. Autom. Reasoning* 58(1):67–95.

Ceylan, İ. İ.; Darwiche, A.; and van den Broeck, G. 2017. Open-world probabilistic databases: An abridged report. In *Proc. IJCAI 2017*, 4796–4800. AAAI Press.

Dalvi, N. N., and Suciu, D. 2007. Efficient query evaluation on probabilistic databases. *VLDB J.* 16(4):523–544.

Dylla, M.; Miliaraki, I.; and Theobald, M. 2013. A temporal-probabilistic database model for information extraction. *Proceedings of the VLDB Endowment* 6(14):1810–1821.

Elkin, P. L.; Brown, S. H.; Husser, C. S.; Bauer, B. A.; Wahner-Roedler, D.; Rosenbloom, S. T.; and Speroff, T. 2006. Evaluation of the content coverage of SNOMED CT: ability of SNOMED clinical terms to represent clinical problem lists. *Mayo Clin. Proc.* 81(6):741–748.

Gabbay, D. M.; Kurucz, A.; Wolter, F.; and Zakharyaschev, M. 2003. *Many-dimensional modal logics: Theory and applications*. Elsevier.

Glimm, B.; Lutz, C.; Horrocks, I.; and Sattler, U. 2008. Conjunctive query answering for the description logic $\mathcal{SHIQ}$. *J. Artif. Intell. Res.* 31:157–204.

Glimm, B.; Horrocks, I.; and Sattler, U. 2008. Unions of conjunctive queries in $\mathcal{SHOQ}$. In *Proc. KR 2008*, 252–262.

Gutiérrez-Basulto, V.; Jung, J. C.; Lutz, C.; and Schröder, L. 2017. Probabilistic description logics for subjective uncertainty. *J. Artif. Intell. Res.* 58:1–66.

Jung, J. C., and Lutz, C. 2012. Ontology-based access to probabilistic data with OWL QL. In *The Semantic Web - ISWC 2012*, volume 7649 of *LNCS*, 182–197. Springer.

Koopmann, P. 2018. Ontology-based query answering for probabilistic temporal data (extended version). LTCS-Report 18-13, Chair for Automata Theory, Institute for Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany. see `https://lat.inf.tu-dresden.de/research/reports.html`.

Kumar, N.; Khunger, M.; Gupta, A.; and Garg, N. 2015. A content analysis of smartphone–based applications for hypertension management. *Journal of the American Society of Hypertension* 9(2):130–136.

Lukasiewicz, T. 2008. Expressive probabilistic description logics. *Artif. Intell.* 172(6-7):852–883.

Lutz, C. 2007. Inverse roles make conjunctive queries hard. In *Proc. DL 2007*, volume 250 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Margara, A.; Urbani, J.; van Harmelen, F.; and Bal, H. 2014. Streaming the web: Reasoning over dynamic data. *Web Semantics: Science, Services and Agents on the World Wide Web* 25:24 – 44.

Ngo, N.; Ortiz, M.; and Simkus, M. 2016. Closed predicates in description logics: Results on combined complexity. In *Proc. KR 2016*, 237–246. AAAI Press.

Ronca, A.; Kaminski, M.; Grau, B. C.; Motik, B.; and Horrocks, I. 2018. Stream reasoning in temporal datalog. In *Proc. AAAI 2018*. AAAI Press.

Toda, S. 1991. PP is as hard as the polynomial-time hierarchy. *SIAM J. Comput.* 20(5):865–877.

van Emde Boas, P. 1997. The convenience of tilings. *Lecture Notes in Pure and Applied Mathematics* 331–363.