News on Temporal Conjunctive Queries

Veronika Thost

Center for Advancing Electronics Dresden (cfaed), TU Dresden
veronika.thost@tu-dresden.de

Abstract. Temporal query languages are important for stream processing, and ontologies for stream reasoning. Temporal conjunctive queries (TCQs) have therefore been investigated recently together with description logic ontologies, and the knowledge we have about the combined complexities is rather complete. However, often the size of the queries and the ontology is negligible, and what costs is the data. We prove a new result on the data complexity of ontology-based TCQ answering and close the gap between co-NP and ExpTime for many description logics.

Keywords: temporal queries, description logics, data complexity

1 Introduction

The temporal nature of data is important in many applications, and the Web offers more and more streaming sources and datasets. Ontologies play an important role in this context: by linking data from heterogeneous sources to the concepts and relations described in an ontology, the integration and automated processing of the data can be considerably enhanced. Queries formulated in the abstract vocabulary of the ontology can then be answered over all the linked datasets.

Medical domain ontologies written in description logics (DLs) may, for example, capture the facts that the varicella zoster virus (VZV) is a virus, that chickenpox is a VZV infection, and that a negative allergy test implies that no allergies are present, by *concept inclusions*: $\mathsf{VZV} \sqsubseteq \mathsf{Virus}$, $\mathsf{Chickenpox} \sqsubseteq \mathsf{VZVInfection}$, $\mathsf{NegAllergyTest} \sqsubseteq \neg\exists\mathsf{AllergyTo}$. Here, Virus is a *concept name* that represents the set of all viruses, and AllergyTo is a *role name*, representing a binary relation connecting patients to allergies; $\exists\mathsf{AllergyTo}$ refers to the domain of this relation. A possible data source storing patient data (e.g., allergy test results and findings) could look as follows:

PID	Name
1	Ann
2	Bob
3	Chris

PID	AllergyTest	Date
1	neg	16.01.2011
2	pos	06.01.1970
3	neg	01.06.2015

PID	Finding	Date
1	Chickenpox	13.08.2007
2	VZV-Infection	22.01.2010
3	VZV-Infection	01.11.2011

The data is then connected to the ontology by mappings [10], which in our example may link the tuple $(1, \text{Chickenpox}, 16.01.2011)$ to the facts $\mathsf{HasFinding}(1, \mathtt{x})$ and $\mathsf{Chickenpox}(\mathtt{x})$. Conceptually, we thus regard a sequence of fact bases, one for each time point we have data for.

Ontology-based query answering (OBQA) over the above knowledge can then, for example, assist in finding appropriate participants for a clinical study, by formulating the eligibility criteria as queries over the—usually linked and heterogeneous—patient data. The following are examples of in- and exclusion conditions for a recently proposed clinical trial:[1] (i) the patient should have been previously infected with VZV or previously vaccinated with VZV vaccine; (ii) the patient should not be allergic to VZV vaccine. We focus on *temporal conjunctive queries* (TCQs), which were originally proposed by [2, 4]. TCQs allow to combine conjunctive queries (CQs) via the Boolean operators and the temporal operators of propositional linear temporal logic LTL [9]. The above criteria can be specified with the following TCQ $\Phi(x)$, to obtain all eligible patients x:

$$\big(\Diamond_P\big(\exists y.\mathsf{HasFinding}(x,y)\wedge\mathsf{VZVInfection}(y)\big)\vee\Diamond_P\big(\exists y.\mathsf{VaccinatedWith}(x,y)\wedge \mathsf{VZVVaccine}(y)\big)\big)\wedge\neg\big(\exists y.\mathsf{AllergyTo}(x,y)\wedge\mathsf{VZVVaccine}(y)\big)$$

We here use the temporal operator "some time in the past" ($\Diamond_P$) and consider the symbols $\mathsf{AllergyTo}$ and $\mathsf{VZVVaccine}$ to be *rigid*, which means that their interpretation does not change over time; that is, we assume someone having an allergy to have this allergy for his whole life. The OBQA scenario outlined above is similar to the classical one [5], but we use temporal queries and consider a (finite) sequence of fact bases. The ontology is written in a classical DL (i.e., one can take one of the many existing ontologies) and assumed to always hold.

In contrast, so-called *temporal DLs* extend classical DLs by temporal operators, which then occur within the ontology (see [11] for an overview). But most of these logics yield high reasoning complexities, even if the underlying atemporal DL allows for tractable reasoning. Lower complexities are only obtained by either considerably restricting the set of temporal operators or the DL.

The combined and data complexity of TCQ entailment have been studied for various DLs in the past [4, 3, 7, 6]. In a nutshell, we have that the combined complexity strongly varies—between PSpace and 2-ExpTime—depending on the DL considered and the rigid names allowed, which often increase complexity.[2] The data complexity for the lightweight DLs between $\textit{DL-Lite}_{core}$ and $\textit{DL-Lite}^{\mathcal{H}}_{horn}$ is generally ALogTime, and the one for $\mathcal{EL}$ without rigid symbols is P, and co-NP with rigid symbols. For all other DLs investigated so far, containment in co-NP has only been shown for the case without rigid roles. This includes expressive DLs such as $\mathcal{SHIQ}$ and is interesting since already standard conjunctive query entailment is co-NP-hard in these DLs, which means that we get the temporal features "for free". However, rigid roles are considered as an important feature for modeling and often expressive DLs are needed; for instance, simple disjunctions of the form $\top \sqsubseteq \mathsf{Male}\sqcup\mathsf{Female}$ ("everyone is male or female") cannot be expressed in $\textit{DL-Lite}^{\mathcal{H}}_{horn}$ or $\mathcal{EL}$. Yet, the proposed algorithms for such combinations are at least exponential in the data.

In this paper, we first close the co-NP/ExpTime gap for the $\textit{DL-Lite}^{\mathcal{H}}_{krom}$—which allows for disjunctions as in the example—and prove that TCQ entailment

[1] https://clinicaltrials.gov/ct2/show/NCT01953900

[2] For some very expressive DLs, we have co-2-NExpTime-hardness/decidability.

is in CO-NP in data complexity, even with rigid roles. Then, we show that this also holds for much more expressive DLs, such as $\mathcal{ALCHI}$.

2 Preliminaries

Description logics focus on *individual names*, which are interpreted as constants; *concepts*, which are interpreted as sets; and *roles*, which are interpreted as binary relations. Accordingly, DL signatures are based on three kinds of symbols: *individual names* $\mathsf{N_I}$, *concept names* $\mathsf{N_C}$, and *role names* $\mathsf{N_R}$, all of which are non-empty, pairwise disjoint sets. We focus on the DL *DL-Lite*$^{\mathcal{H}}_{krom}$ [1].

***DL-Lite*$^{\mathcal{H}}_{krom}$.** Let $a, b \in \mathsf{N_I}$, $A \in \mathsf{N_C}$, and $P \in \mathsf{N_R}$. In *DL-Lite*$^{\mathcal{H}}_{krom}$, the sets of *roles*, *basic concepts*, and *concepts* are defined as follows:

$$R, S ::= P \mid P^-, \qquad B, C ::= \top \mid A \mid \exists R, \qquad D ::= B \mid \neg B$$

where $\cdot^-$ denotes the inverse role operator.

DL-Lite$^{\mathcal{H}}_{krom}$ *axioms* are the following kinds of expressions: *concept inclusions (CIs)* are of the form $B \sqsubseteq C$, $B \sqsubseteq \neg C$, or $\neg B \sqsubseteq C$; *role inclusions (RIs)* of the form $R \sqsubseteq S$; and *assertions* of the form $B(a), \neg B(a), P(a, b)$, or $\neg P(a, b)$.

A *DL-Lite*$^{\mathcal{H}}_{krom}$ *ontology* is a finite set of concept and role inclusions, and an *ABox* is a finite set of assertions. Together, an ontology $\mathcal{O}$ and an ABox $\mathcal{A}$ form a *knowledge base* (KB) $\mathcal{K} := \mathcal{O} \cup \mathcal{A}$, written $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$.

We sometimes also refer to the ABox as *fact base* or simply as the *data*. Without loss of generality, we assume that, if the RI $R \sqsubseteq S$ is contained in $\mathcal{O}$, then we also have $\exists R \sqsubseteq \exists S \in \mathcal{O}$ and $\exists R^- \sqsubseteq \exists S^- \in \mathcal{O}$; and that $\mathcal{O}$ contains the trivial axioms $\exists R \sqsubseteq \exists R$ for all roles R occurring in $\mathcal{O}$. The set of roles is denoted by $\mathsf{N_R^-}$. For a given KB $\mathcal{K} := \mathcal{O} \cup \mathcal{A}$, we denote by $\mathsf{N_I}(\mathcal{K})$ and $\mathsf{N_I}(\mathcal{A})$ the set of individual names that occur in $\mathcal{K}$ and $\mathcal{A}$; by $\mathsf{N_C}(\mathcal{O})$ and $\mathsf{N_R}(\mathcal{O})$ the sets of concept and role names occurring in $\mathcal{K}$; and by $\mathsf{N_R^-}(\mathcal{O})$ the set of roles occurring in $\mathcal{K}$. $\mathbb{B}(\mathcal{O})$ and $\mathbb{C}(\mathcal{O})$ denote the sets of all basic concepts and, respectively, concepts that can be built from the symbols in $\mathsf{N_C}(\mathcal{O})$ and $\mathsf{N_R^-}(\mathcal{O})$. We may also use the abbreviation $(P^-)^- := P$ for $P \in \mathsf{N_R}$.

A DL *interpretation* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ consists of a non-empty set $\Delta^\mathcal{I}$, the *domain* of $\mathcal{I}$, and an *interpretation function* $\cdot^\mathcal{I}$, which assigns to every $A \in \mathsf{N_C}$ a set $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$, to every $P \in \mathsf{N_R}$ a binary relation $P^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$, and to every $a \in \mathsf{N_I}$ an element $a^\mathcal{I} \in \Delta^\mathcal{I}$ such that, for all $a, b \in \mathsf{N_I}$ with $a \neq b$, we have $a^\mathcal{I} \neq b^\mathcal{I}$ (unique name assumption). The function is extended to all roles and concepts: $P^- = \{(y, x) \mid (x, y) \in P\}$, $\top^\mathcal{I} = \Delta^\mathcal{I}$, $\exists R^\mathcal{I} = \{x \in \Delta^\mathcal{I} \mid \exists y \in \Delta^\mathcal{I}, (x, y) \in R^\mathcal{I}\}$, $(\neg D)^\mathcal{I} = \Delta^\mathcal{I} \setminus D^\mathcal{I}$. An interpretation $\mathcal{I}$ *satisfies* (or is a *model* of) an axiom α, written $\mathcal{I} \models \alpha$, if: $\alpha = X \sqsubseteq Y$ is a CI or RI, and $X^\mathcal{I} \subseteq Y^\mathcal{I}$; $\alpha = (\neg)B(a)$ and $a^\mathcal{I} \in B^\mathcal{I}$ ($a^\mathcal{I} \notin B^\mathcal{I}$); $\alpha = (\neg)P(a, b)$ and $(a^\mathcal{I}, b^\mathcal{I}) \in P^\mathcal{I}$ ($(a^\mathcal{I}, b^\mathcal{I}) \notin P^\mathcal{I}$). $\mathcal{I}$ *satisfies* (or is a *model* of) a KB $\mathcal{K}$, written $\mathcal{I} \models \mathcal{K}$, if it satisfies all axioms contained in it. A KB $\mathcal{K}$ is *consistent* (or *satisfiable*) if it has a model, and it is *inconsistent* (or *unsatisfiable*) otherwise. $\mathcal{K}$ *entails* an axiom α, written $\mathcal{K} \models \alpha$, if all models of $\mathcal{K}$ also satisfy α. This terminology and notation is extended to (single) axioms, ontologies, and ABoxes by regarding each as a (singleton) KB. We denote non-entailment by $\mathcal{K} \not\models \alpha$.

In the temporal setting, we assume that some concept and role names are designated as being *rigid* (vs. *flexible*) as outlined in Section 1. If a concept (axiom) contains only rigid symbols, then we may call it a *rigid* concept (axiom). We denote by $\mathsf{N_{RC}} \subseteq \mathsf{N_C}$ the rigid concept and by $\mathsf{N_{RR}} \subseteq \mathsf{N_R}$ the rigid role names.

Temporal Semantics. An infinite sequence $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ of interpretations $\mathcal{I}_i = (\Delta, \cdot^{\mathcal{I}_i})$ is a *DL-LTL structure* if it *respects rigid names*, that is: $X^{\mathcal{I}_i} = X^{\mathcal{I}_j}$ for all $X \in \mathsf{N_I} \cup \mathsf{N_{RC}} \cup \mathsf{N_{RR}}$ and $i, j \geq 0$. Observe that the interpretations in a DL-LTL structure share one domain (constant domain assumption). We may use that terminology in other settings in that we consider interpretations $\mathcal{I}_1, \dots, \mathcal{I}_\ell$ to *respect rigid names* if they agree on the interpretation of all rigid symbols.

Temporal Knowledge Bases. A *temporal knowledge base* (TKB) is of the form $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0\leq i\leq n}\rangle$ with an ontology $\mathcal{O}$ and a non-empty, finite *sequence of ABoxes*. We assume all concept and role names occurring in some ABox of a TKB to also occur in its ontology. $\mathsf{N_I}(\mathcal{K})$ denotes the set of all individual names occurring in the TKB $\mathcal{K}$. Note that every KB can be regarded as a TKB with an ABox sequence of length one.

A DL-LTL structure $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ over a domain Δ is a *model* of a TKB $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0\leq i\leq n}\rangle$, written $\mathfrak{I} \models \mathcal{K}$, if $\mathcal{I}_i \models \mathcal{O}$ for all $i \geq 0$ and $\mathcal{I}_i \models \mathcal{A}_i$ for all $i \in [0, n]$. A TKB is *consistent* (or *satisfiable*) if it has a model, and it is *inconsistent* (or *unsatisfiable*) otherwise.

Temporal Conjunctive Queries. Let $\mathsf{N_V}$ be the set of variables, and $\mathsf{N_T} := \mathsf{N_I} \cup \mathsf{N_V}$ be the set of terms. A *conjunctive query* (CQ) is of the form $\exists y_1, \dots, y_m.\psi$, where $y_1, \dots, y_m \in \mathsf{N_V}$ and ψ is a (possibly empty) finite conjunction of *concept atoms* of the form $A(t)$ and *role atoms* of the form $R(s, t)$, where $A \in \mathsf{N_C}$, $R \in \mathsf{N_R}$ and $s, t \in \mathsf{N_T}$. The set of *temporal conjunctive queries* (TCQs) is defined as follows, where φ is a CQ:

$$\Phi, \Psi ::= \varphi \mid \neg\Phi \mid \Phi \wedge \Psi \mid \bigcirc_F \Phi \mid \bigcirc_P \Phi \mid \Phi\,\mathcal{U}\,\Psi \mid \Phi\,\mathcal{S}\,\Psi$$

A TCQ Φ is a *CQ literal* if it is of the form $(\neg)\varphi$ with φ being a CQ; it is *positive* if $\Phi = \varphi$, and otherwise *negative*.

We denote the set of individuals occurring in a TCQ Φ by $\mathsf{N_I}(\Phi)$. As in propositional LTL, we may use abbreviations *true*[3] and *false*. The empty conjunction and disjunction are interpreted as *true* and *false*, respectively.

As usual, the semantics is defined in a model-theoretic way, based on the notion of homomorphisms. A mapping $\pi\colon \mathsf{N_T}(\varphi) \to \Delta^{\mathcal{I}}$ is a homomorphism of a CQ φ into an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ if $\pi(a) = a^{\mathcal{I}}$ for all $a \in \mathsf{N_I}(\varphi)$, $\pi(t) \in A^{\mathcal{I}}$ for all concept atoms $A(t)$ in φ, and $(\pi(s), \pi(t)) \in R^{\mathcal{I}}$ for all role atoms $R(s, t)$ in φ. $\mathcal{I}$ *satisfies* (or is a *model* of) φ, written $\mathcal{I} \models \varphi$, if there is such a homomorphism. For a given DL-LTL structure $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$, an $i \geq 0$, and TCQ Φ, the satisfaction relation $\mathfrak{I}, i \models \Phi$ is defined by induction on the

[3] For instance, *true* may denote a fix TCQ $\varphi \vee \neg\varphi$, where φ is an arbitrary CQ.

TCQ Φ'	Condition for $\mathfrak{I}, i \models \Phi'$
CQ φ	$\mathcal{I}_i \models \varphi$
$\neg\Phi$	$\mathfrak{W}, i \not\models \Phi$
$\Phi \wedge \Psi$	$\mathfrak{I}, i \models \Phi$ and $\mathfrak{I}, i \models \Psi$
$\bigcirc_F \Phi$	$\mathfrak{I}, i+1 \models \Phi$
$\bigcirc_P \Phi$	$i > 0$ and $\mathfrak{I}, i-1 \models \Phi$
$\Phi \,\mathcal{U}\, \Psi$	there is a $k \geq i$, such that $\mathfrak{I}, k \models \Psi$ and, for all j, $i \leq j < k$, we have $\mathfrak{I}, j \models \Phi$
$\Phi \,\mathcal{S}\, \Psi$	there is a k, $0 \leq k \leq i$, such that $\mathfrak{I}, k \models \Psi$ and, for all j, $k < j \leq i$, we have $\mathfrak{I}, j \models \Phi$

Fig. 1. Semantics of TCQs given a DL-LTL structure $\mathfrak{I} = (\mathcal{I}_i)_{i \geq 0}$.

structure of Φ as specified in Figure 1. $\mathfrak{I}$ is a *model* of Φ w.r.t. a TKB $\mathcal{K}$ if $\mathfrak{I} \models \mathcal{K}$ and $\mathfrak{I}, n \models \Phi$. A TCQ Φ is *satisfiable* w.r.t. a TKB $\mathcal{K}$ if it has a model w.r.t. $\mathcal{K}$; and Φ is *entailed* by a TKB $\mathcal{K}$, written $\mathcal{K} \models \Phi$, if every model of $\mathcal{K}$ is also a model of Φ w.r.t. $\mathcal{K}$. We denote the fact that $\mathfrak{I}, i \models \Phi$ and $\mathcal{K} \models \Phi$ do not hold by $\mathfrak{I}, i \not\models \Phi$ and $\mathcal{K} \not\models \Phi$. Observe that a model of a TCQ must satisfy the query at the current time point n, which is different for propositional LTL if $n > 0$.

Without loss of generality, we assume that the CQs contained in a TCQ Φ use disjoint variables and denote by $\mathcal{Q}_\Phi$ the set of exactly those CQs.[4] We further assume that TCQs contain only individual names that occur in the ABoxes, and only concept and role names that occur in the ontology, and that all CQs contained in TCQs are *connected* (i.e., the corresponding Gaifman graph is connected); it is easy to show that this is without loss of generality.

Solving TCQ Satisfiability. The TCQ satisfiability problem can be split into two separate ones: one in propositional LTL and one or several "atemporal" ones in DL [4, Lemma 4.7]. The former tests the satisfiability of the propositional abstraction of the given TCQ Φ at n, which is obtained from Φ by replacing the CQs $\varphi_1, \dots, \varphi_m \in \mathcal{Q}_\Phi$ by propositional variables $p_1, \dots, p_m$, respectively. The idea is that the worlds $w_0, w_1, \dots$ in the LTL model characterize the satisfaction of the CQs from $\mathcal{Q}_\Phi$ in the respective DL interpretations $\mathcal{I}_0, \mathcal{I}_1, \dots$ such that, to obtain $\mathcal{I}_i$, we only have to check the satisfiability of the conjunction of CQ literals induced by w_i w.r.t. the atemporal KB $\langle \mathcal{O}, \mathcal{A}_i \rangle$, where $\mathcal{A}_i = \emptyset$ for $i > n$. From the latter it can be seen that, assuming k to be the number of different worlds occurring in the LTL model, it is sufficient to look for $n + 1 + k$ corresponding DL interpretations. More precisely, the problems are linked by a set $\mathcal{W} = \{W_1, \dots, W_k\} \subseteq 2^{\{p_1, \dots, p_m\}}$, which collects all worlds occurring in the LTL model, and a mapping $\iota\colon [0, n] \to [1, k]$ that maps time points to indexes from $\mathcal{W}$ and points out the first $n + 1$ worlds, which have to reflect the knowledge given in the respective ABoxes. The DL part is defined as *r-satisfiability*; the set $\mathcal{W}$ is *r-satisfiable* w.r.t. ι and a TKB $\mathcal{K}$ iff there are interpretations $\mathcal{I}_0, \dots, \mathcal{I}_n$, $\mathcal{J}_1, \dots, \mathcal{J}_k$ as follows:

[4] If the variables were not disjoint, we could simply rename them.

– the interpretations share the same domain and respect rigid names,
– the interpretations are models of $\mathcal{O}$,
– $\mathcal{J}_i$ is a model of $\chi_i := \bigwedge_{p_j \in W_i} \varphi_j \wedge \bigwedge_{p_j \in \overline{W_i}} \neg\varphi_j$ for all $i \in [1, k]$,
– $\mathcal{I}_i$ is a model of $\mathcal{A}_i$ and $\chi_{\iota(i)}$ for all $i \in [0, n]$.

Observe that, regarding data complexity, $\mathcal{W}$ and ι can be guessed in constant and linear time, respectively. [4, Lem. 4.12] show that the LTL satisfiability problem w.r.t. a given $\mathcal{W}$ and ι can be decided in polynomial time. However, regarding r-satisfiability, [4] only show membership in ExpTime. The critical point with r-satisfiability is the requirement that the interpretations for the $n + k + 1$ relevant time points share a common domain, so the individual satisfiability tests have to be done together. The trivial approach is to rename the flexible names for all $i \in [0, n + k]$. This requires however that the ontology is extended by corresponding axioms; that is, it grows with the data and impacts complexity.

3 Characterizing r-Satisfiablility

We regard a TCQ Φ, a TKB $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0 \leq i \leq n} \rangle$ in *DL-Lite*$^{\mathcal{H}}_{krom}$, a set $\mathcal{W} \subseteq 2^{\{p_1,\ldots,p_m\}}$ such that $\mathcal{W} = \{W_1, \ldots, W_k\}$, and a mapping $\iota \colon [0, n] \to [1, k]$, as described in the previous section. The goal is to propose a characterization of r-satisfiability of $\mathcal{W}$ w.r.t. ι and $\mathcal{K}$ which, in contrast to existing characterizations, is tailored to *DL-Lite*$^{\mathcal{H}}_{krom}$ and shows that the r-satisfiability problem is in NP.

Observe that the functions of the shared domain in the definition of r-satisfiability are mainly two: (i) to synchronize the interpretation of rigid symbols regarding the named individuals; (ii) to guarantee that the satisfiability of the conjunctions χ_i, $i \in [1, k]$, which is represented by the respective interpretation $\mathcal{J}_i$, is not contradicted by the interpretation of the rigid names in the other interpretations, especially in no $\mathcal{I}_i$ with $i \in [0, n]$. The idea is to look for similar interpretations $\mathcal{I}_0, \ldots, \mathcal{I}_n, \mathcal{J}_1, \ldots, \mathcal{J}_k$, but to not require a shared domain: based on $\mathcal{K}$, $\mathcal{W}$, and ι, we specify a polynomial amount of additional data—that hence can be guessed in polynomial time—, which captures knowledge restricting the interpretation of the individual and rigid names and simulates the shared domain; the additional data then allows us to check the conditions for r-satisfiability for each of the interpretations independently of the other interpretations—nondeterministically, in polynomial time. Without loss of generality, we can restrict our focus to certain *canonical interpretations*, based on the standard chase [8]; we introduce elements of the form $u_{aR_1\ldots R_\ell}$, $a \in \mathsf{N_I}$, $R_1, \ldots, R_\ell \in \mathsf{N_R^-}$. Note that we also apply this general approach for *DL-Lite*$^{\mathcal{H}}_{horn}$ and $\mathcal{EL}$ in [6, 7], but we do not have to deal with nondeterminism there; that is, there is only one canonical interpretation for a KB.

Definition 1 (Canonical Interpretation). *Let $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$ be a consistent DL-Lite$^{\mathcal{H}}_{krom}$ knowledge base. For all $A \in \mathsf{N_C}$ and $P \in \mathsf{N_R}$, define:*

$$A^0 := \{a \mid A(a) \in \mathcal{A}\},$$
$$P^0 := \{(a, b) \mid P(a, b) \in \mathcal{A}\} \cup \{(a, u_{aP}) \mid \exists P(a) \in \mathcal{A}\} \cup \{(u_{aP^-}, a) \mid \exists P^-(a) \in \mathcal{A}\}.$$

Then, iterate over all $i \geq 0$*: for all* $X \in \mathsf{N_C} \cup \mathsf{N_R}$ *define* $X^{i+1} := X^i$*; apply one of the following rules for all* $A \in \mathsf{N_C}$*,* $P \in \mathsf{N_R}$*,* $R, S \in \mathsf{N_R^-}$*, and* $B, C \in \mathbb{B}(\mathcal{O})$*; and increment* i*;* $(d,e) \in (P^-)^i$ *denotes the fact that* $(e,d) \in P^i$*, and* $d \in (\exists R)^i$ *denotes the existence of an element* e *such that* $(d,e) \in R^i$*:*

- *If* $B \sqsubseteq A \in \mathcal{O}$ *and* $e \in B^i$*, then add* e *to* A^{i+1}*.*
- *If* $B \sqsubseteq \exists R \in \mathcal{O}$ *and* $e \in B^i$*:*
  - *if* $e \in \mathsf{N_I}(\mathcal{A})$*, then add* (e, u_{eR}) *to* R^{i+1}*;*
  - *if* $e = u_\varrho$*, then add* $(e, u_{\varrho R})$ *to* R^{i+1}*.*
- *If* $\exists R \sqsubseteq A \in \mathcal{O}$*,* $(d,e) \in R^i$*, then add* d *to* A^{i+1}*.*
- *If* $R \sqsubseteq S \in \mathcal{O}$ *and* $(d,e) \in R^i$*, then add* (d,e) *to* S^{i+1}*.*
- *If* $\neg B \sqsubseteq C \in \mathcal{O}$*,* $e \notin B^i$*,* $e \notin C^i$*, and every other rule (i.e., for a CI without negation) that applies to* e *or a tuple containing* e *has been applied in a step* $j < i$*, then add* e *to* B^{i+1} *or* C^{i+1}*.*

The set $\Delta_{\mathsf{u}}^{\mathcal{I}_\mathcal{K}}$ *collects the above introduced new elements.*

A canonical interpretation $\mathcal{I}_\mathcal{K}$ *for* $\mathcal{K}$ *is then defined as follows based on such a sequence of rule applications, for all* $a \in \mathsf{N_I}(\mathcal{A})$*,* $A \in \mathsf{N_C}$*, and* $P \in \mathsf{N_R}$*:*

$$\Delta^{\mathcal{I}_\mathcal{K}} := \mathsf{N_I}(\mathcal{A}) \cup \Delta_{\mathsf{u}}^{\mathcal{I}_\mathcal{K}}, \quad a^{\mathcal{I}_\mathcal{K}} := a, \quad A^{\mathcal{I}_\mathcal{K}} := \bigcup_{i=0}^{\infty} A^i, \quad P^{\mathcal{I}_\mathcal{K}} := \bigcup_{i=0}^{\infty} P^i.$$

Note that the assumptions in Section 2 about the additional axioms in the ontology ensure that, whenever there is a named individual $a \in (\exists R)^i$ for some $i \geq 0$, then a has an R-successor of the form u_{aR} in the corresponding canonical interpretation, and similar for unnamed elements. We denote the restriction of a canonical interpretation $\mathcal{I}$ to a named individual a and its unnamed successors by $\mathcal{I}_{|a}$. If $\mathcal{K}$ is consistent, then there is a canonical interpretation for $\mathcal{K}$ that is a model of $\mathcal{K}$. We denote the set of all those canonical models by $\mathbb{I}_\mathcal{K}$.

In what follows, we specify the additional data to overtake the Functions (i) and (ii). Specifically, we define a set of ABoxes containing assertions that (i) (largely) fix the interpretations on the named individuals (i.e., relations to unnamed successors are not fully taken into account yet), and (ii) ensure both that the positive CQ literals are satisfied as required and that the negative CQ literals are not satisfied if they must not. For simplicity, for $i \in [1,k]$, we define $\mathcal{A}_{n+i} := \emptyset$ and extend ι such that $\iota(n+i) := i$. For the synchronization of the named individuals, we use *name-ABoxes*, which are similar to the ABox types defined for *DL-Lite*$_{horn}^{\mathcal{H}}$ in [6], but we include flexible symbols. The idea is to then guess $n+k+1$ name-ABoxes and require them to agree on the rigid assertions.

Definition 2 (Name-ABox). *A* name-ABox *for a set of individual names* I *w.r.t.* $\mathcal{O}$ *is a set* $\mathcal{A}$ *of assertions formulated over* I *and all symbols in* $\mathbb{B}(\mathcal{O})$ *and* $\mathsf{N_R}(\mathcal{O})$ *such that* $\alpha \in \mathcal{A}$ *iff* $\neg\alpha \notin \mathcal{A}$*.*

Second, for all $i \in [0, n+k]$, define $Q_i := \{\varphi_j \mid p_j \in W_{\iota(i)}\}$. Let the set $\mathsf{N_I^{aux}} \subseteq \mathsf{N_I}$ contain an individual name a_x^i for each $i \in [0, n+k]$ and each variable x occurring in a CQ in Q_i. Note that, because of our assumption that the

CQs in Φ have no variables in common, each $a_x^i \in \mathsf{N}_\mathsf{I}^{\mathsf{aux}}$ can be unambiguously associated to a CQ containing x. Then, $\mathcal{A}_{Q_i}$ denotes the ABox obtained from Q_i by instantiating all variables with the corresponding names from $\mathsf{N}_\mathsf{I}^{\mathsf{aux}}$. We use these ABoxes to ensure that the positive CQ literals are satisfied as required.

While the former is similarly done in [6, 7], the nondeterminism allowed in $DL\text{-}Lite_{krom}^{\mathcal{H}}$ requires a more careful construction of the unnamed parts of $\mathcal{I}_0, \dots, \mathcal{I}_n$, $\mathcal{J}_1, \dots, \mathcal{J}_k$ (i.e., since they have to satisfy all CIs of the form $\neg B \sqsubseteq C$ in $\mathcal{O}$, we have to specify them correspondingly): we must ensure that the interactions of $\mathcal{I}_0, \dots, \mathcal{I}_n$, $\mathcal{J}_1, \dots, \mathcal{J}_k$ in those parts, which are caused by the rigid names, do not lead to the satisfaction of some $\varphi_j \in \mathcal{Q}_\Phi$ in some $\mathcal{J}_i$ ($\mathcal{I}_i$) although we have that $p_j \in \overline{W_i}$ ($p_j \in \overline{W_{\iota(i)}}$). The idea for the construction of $\mathcal{I}_0, \dots, \mathcal{I}_n$, $\mathcal{J}_1, \dots, \mathcal{J}_k$ is to not consider arbitrary trees of unnamed successors for all the individuals in all the interpretations, but to define prototypical ones whose size is constant in the data, that fix the interpretations, and which we then copy for all named individuals that are sufficiently similar. To this end, we define *types*, which are generally independent of the data; for every interpretation and individual name, there is however exactly one type characterizing the former on the latter. A type captures the basic concepts satisfied on a name and, in particular, relevant homomorphisms of CQs from $\mathcal{Q}_\Phi$ w.r.t. the named individual and its unnamed successors; in particular, it does not explicitly refer to individual names. A *temporal type* is a set of types. The idea is to consider prototypical trees of unnamed successors for each temporal type as additional data: we use a set of prototypical *tree-ABoxes* (one per type) over the same names, which agree on the interpretation of the rigid names, and are such that every ABox represents some interpretation on the unnamed successors that fits to the respective type. For instance, if a type specifies a CQ $\varphi \in \mathcal{Q}_\Phi$ to be *not* satisfied (w.r.t. the unnamed successors), then φ is not satisfied in the ABox. Our main contribution is that we show that such tree-ABoxes whose size is independent of the data do exist in the case of r-satisfiability and that we can assemble the interpretations $\mathcal{I}_0, \dots, \mathcal{I}_n$, $\mathcal{J}_1, \dots, \mathcal{J}_k$ from these ABoxes: for every individual name a and all $i \in [0, n+k]$, we guess a type $\mathcal{T}_{a,i}$—a polynomial amount of information—that represents $\mathcal{I}_i$ (or $\mathcal{J}_i$) on a; the set of all these types for a is a temporal type and yields the prototypical successors to choose; that is, we copy the elements in the corresponding set of tree-ABoxes and then specify $\mathcal{I}_i$ (or $\mathcal{J}_i$) on these elements according to the ABox for $\mathcal{T}_{a,i}$. Observe that we use finite ABoxes, which means that every of the ABoxes contains enough information to define the interpretations on other required successors (i.e., we may copy the elements several times). Since the ABoxes capture all the rigid information from other time points, this allows us to test the satisfaction of the negative CQ literals for every of the $n+k+1$ interpretations individually.

Definition 3 (Type). *A* basic type *is a set $\mathcal{B} \subseteq \mathbb{C}(\mathcal{O})$ such that $B \in \mathcal{B}$ iff $\neg B \notin \mathcal{B}$ for all $B \in \mathbb{B}(\mathcal{O})$; given such a basic type, the corresponding set of assertions is defined as $\mathcal{A}_\mathcal{B}(a) := \{D(a) \mid D \in \mathcal{B}\}$. The basic type of an individual name a in an interpretation $\mathcal{I}$ is the set $\mathsf{BT}(a, \mathcal{I}) := \{D \in \mathbb{C}(\mathcal{O}) \mid a \in D^\mathcal{I}\}$.*

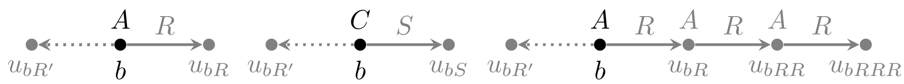
A type *is a triple* $(\mathcal{B}, \mathcal{M}, \mathcal{Q})$ *with a basic type* $\mathcal{B}$*, a set* $\mathcal{M} \subseteq \bigcup_{\varphi \in \mathcal{Q}_\Phi} 2^{\mathsf{N_T}(\varphi)}$ *of term sets, and a set* $\mathcal{Q} \subseteq \mathcal{Q}_\Phi$ *of CQs. The* type *of an individual name* a *in a canonical interpretation* $\mathcal{I}$ *is the triple* $\mathsf{T}(a, \mathcal{I}) := (\mathsf{BT}(a, \mathcal{I}), \mathcal{M}, \mathcal{Q})$ *where* $\mathcal{Q} \subseteq \mathcal{Q}_\Phi$ *contains exactly the CQs that are satisfied in* $\mathcal{I}_{|a}$*, and* $\mathcal{M}$ *contains all sets* S *of terms for which there are a CQ* $\varphi \in \mathcal{Q}_\Phi$ *and a partial homomorphism* $\pi \colon \mathsf{N_T}(\varphi) \to \Delta^{\mathcal{I}_{|a}}$ *of* φ *into* $\mathcal{I}_{|a}$ *with* $a \in \mathsf{range}(\pi)$ *and* $\mathsf{dom}(\pi) = S$*.*

A temporal type *is a set of types.*

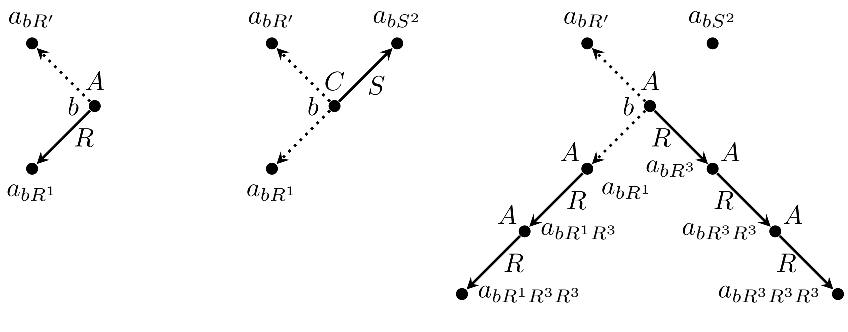
We assume every temporal type τ to be an ordered set and use τ_i to refer to the i-th type in τ. We denote the set of all temporal types by $\mathbb{T}$.

It is left to specify the prototypical tree ABoxes of unnamed successors for a given temporal type τ. We first construct ABoxes for the types and, in a second step, ensure that the conditions specified by the types are satisfied in them, respectively. These ABoxes $(\mathcal{A}_{\tau_i})_{1 \le i \le |\tau|}$, initially empty, are constructed iteratively, based on canonical interpretations, amongst others for these ABoxes. During the iteration, we therefore assume that these interpretations are (non-deterministically) extended correspondingly (i.e., to cover the new elements of the ABoxes). Let $s = |\tau|$. For all $i \in [1, s]$, consider some $\mathcal{I}_{\tau_i} \in \mathbb{I}_{\langle \mathcal{O}, \mathcal{A}_{\tau_i} \cup \mathcal{A}_{\mathcal{B}_i}(b) \rangle}$; b is a fresh individual name and $\mathcal{B}_i$ the basic type in τ_i. Our procedure takes the sequence $(\mathcal{I}_{\tau_i})_{1 \le i \le s}$ as input. It then repeatedly iterates over the (extended) interpretations and extends the ABoxes $(\mathcal{A}_{\tau_i})_{1 \le i \le |\tau|}$ until nothing changes any more.

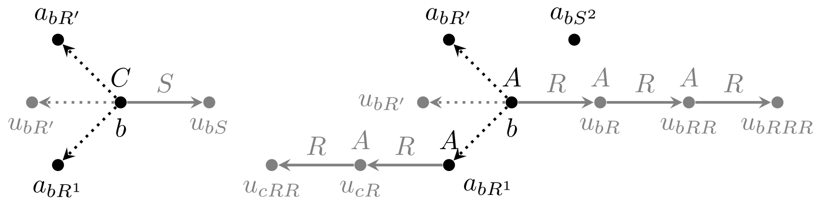
Example 1. We consider an ontology containing the inclusions $\neg A \sqsubseteq B, \neg B \sqsubseteq C$, $A \sqsubseteq \exists R, C \sqsubseteq \exists S, R \sqsubseteq R'$, where only R' is rigid. Let further $|\tau| = 3$ and the input canonical interpretations for τ_1, τ_2, τ_3 be as follows.



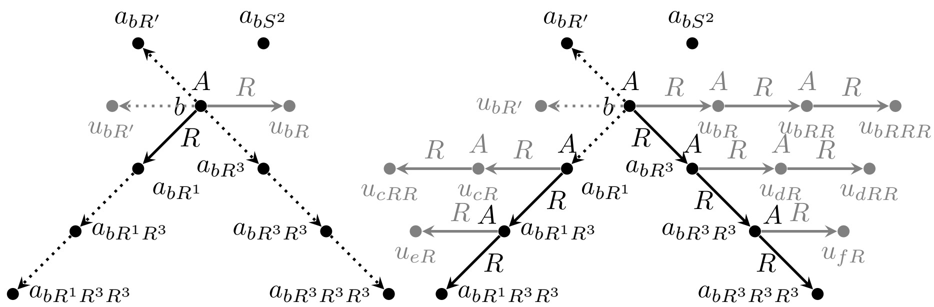
Note that all elements that are no instances of A or C instantiate B, R' is dotted. After one iteration over the interpretations, the ABoxes $\mathcal{A}_{\tau_i}$ for $i \in [1, 3]$ are:



$\mathcal{A}_{\tau_1}$ is obtained from $\mathcal{I}_{\tau_1}$ by introducing names representing the unnamed elements, flexible roles get a superscript. All names and the rigid assertions from $\mathcal{A}_{\tau_1}$ are then added to the other ABoxes, and the interpretations $\mathcal{I}_{\tau_2}$ and $\mathcal{I}_{\tau_3}$ are extended correspondingly. The above $\mathcal{A}_{\tau_2}$ is then obtained from this $\mathcal{I}_{\tau_2}$, assumed to be as below. Then, again, all names and rigid assertions from $\mathcal{A}_{\tau_2}$ are added to the other ABoxes, the interpretations are extended, and the above $\mathcal{A}_{\tau_3}$ is obtained from the extended $\mathcal{I}_{\tau_3}$, depicted below, where $c = a_{bR^1}$.



Note that we do not depict all R'-successors; according to Definition 1, all elements instantiating $\exists R$ must have such successors. We lastly show $\mathcal{I}_{\tau_1}$ and $\mathcal{I}_{\tau_3}$ extended for the above $\mathcal{A}_{\tau_3}$, where $d = a_{bR^3}$, $e = a_{bR^1R^3}$, and $f = a_{bR^3R^3}$. Observe that we assume that $\mathcal{I}_{\tau_3}$ interprets u_{dR} in the same way as u_{bRR}, both u_{dRR} and u_{fR} according to u_{bRRR}, and u_{eR} according to u_{cRR}.



In order to ensure that the size of the tree-ABoxes is finite, we specify a termination criterion based on the maximal size $m := \max\{|\varphi| \mid \varphi \in \mathcal{Q}_\Phi\}$ of a single of the CQs: we stop the introduction of new elements $a_{\varrho\varrho'\omega R}$ with $|\varrho|, |\omega| > m$ if those would, thereafter, occur in a subtree (of depth $> m$) with root $a_{\varrho\varrho'}$ that would be a copy of an already existing subtree of depth m with root a_ϱ. This approach is correct if we extend the ABoxes in a breadth-first fashion and, especially, regard all of the canonical interpretations before extending the trees one level deeper.

We now specify the procedure TreeABox, which takes τ, the interpretation sequence $(\mathcal{I}_{\tau_i})_{1\leq i\leq s}$, and b as input:

- For each domain element $u_{b\varrho R}$ of $\mathcal{I}_{\tau_i}$, introduce an individual name $a_{b\varrho R}$ if $R \in \mathsf{N_{RR}}$, and otherwise $a_{b\varrho R^i}$; we assume that such individual names and role names containing superscripts do not occur in $\mathcal{K}$. Similarly, for each domain element $u_{c\varrho R}$ of $\mathcal{I}_{\tau_i}$ such that $c = a_{b\sigma} \in \mathsf{N_I^{tree}}$, introduce a new individual name $a_{b\sigma\varrho R}$ if $R \in \mathsf{N_{RR}}$ and otherwise $a_{b\sigma\varrho R^i}$.
  Let $\mathcal{R} := \mathsf{N_{RR}^-} \cup \{R_i \mid R \in \mathsf{N_R^-} \setminus \mathsf{N_{RR}^-}, 1 \leq i \leq s\}$. The set $\mathsf{N_I^{tree}}$ collects the new individual names, but (*) a name a_υ is only added if there are no names

a_ϱ, and a_σ in $\mathcal{A}_\tau$ such that $\sigma = \varrho\varrho'$; $\upsilon = \sigma\sigma'$ with $|\sigma'| > m$; and, for all $\omega \in \mathcal{R}^m$, $a_{\varrho\omega} \in \mathsf{N}_\mathsf{I}^{\mathsf{tree}}$ iff $a_{\sigma\omega} \in \mathsf{N}_\mathsf{I}^{\mathsf{tree}}$, and $a_{\varrho\omega}$ and $a_{\sigma\omega}$ have the same basic type in any of the interpretations $\mathcal{I}_{\tau_\ell}$, $\ell \in [1, s]$. For the next step, we capture this relation using the function $\nu\colon \Delta_\mathsf{u}^{\mathcal{I}_{\tau_i}} \to \mathsf{N}_\mathsf{I}^{\mathsf{tree}}$: for the above elements $u_{b\varrho}$, define $\nu(u_{b\varrho R}) := a_{b\varrho R^{(i)}}$ and, for the others, define $\nu(u_{c\varrho R}) := a_{b\sigma\varrho R^{(i)}}$. Observe that this function may map several elements to the same name; as it is the case for u_{bRR} and u_{dR} in Example 1, where we have $\nu(u_{bRR}) = a_{bR^3R^3}$ and $\nu(u_{dR}) = a_{bR^3R^3}$. In these cases, we assume that the unnamed successors of elements from $\mathsf{N}_\mathsf{I}^{\mathsf{tree}}$, such as u_{dR}, are interpreted in the same way as the original unnamed elements, such as u_{bRR}, for which ν was defined first (i.e., this must have happened when a was introduced, hence u_{dR} did not yet exist); and that the successors of the former are interpreted in the same way as the corresponding successors of the latter, and so on.

- For every $a \in \mathsf{N}_\mathsf{I}^{\mathsf{tree}}$ introduced in the previous step, let $u_{\sigma R} \in \Delta_\mathsf{u}^{\mathcal{I}_{\tau_i}}$ be one of the elements for which a was created. Add the following assertions to $\mathcal{A}_{\tau_i}$:
  - for every $B \in \mathbb{B}(\mathcal{O})$ such that $u_{\sigma R} \in B^{\mathcal{I}_{\tau_i}}$, the assertion $B(a)$;
  - for every $S \in \mathsf{N}_\mathsf{R}^-$ such that $(\sigma, u_{\sigma R}) \in S^{\mathcal{I}_{\tau_i}}$, the assertion $S(\sigma, a)$;
  - for every $S \in \mathsf{N}_\mathsf{R}^-$ such that $(u_\sigma, u_{\sigma R}) \in S^{\mathcal{I}_{\tau_i}}$, the assertion $S(\nu(u_\sigma), a)$.

  Add all of the individual names and the rigid assertions added to $\mathcal{A}_{\tau_i}$ also to all other $\mathcal{A}_{\tau_j}$, $j \in [1, s]$.[5]
- For every $a \in \mathsf{N}_\mathsf{I}^{\mathsf{tree}}$ and $B \in \mathbb{B}(\mathcal{O})$ such that $a \in B^{\mathcal{I}_{\tau_i}}$, further add the assertion $B(a)$ to $\mathcal{A}_{\tau_i}$. Again, add the rigid assertions to all $\mathcal{A}_{\tau_j}$, $j \in [1, s]$.

The procedure outputs the sequence $(\mathcal{A}_{\tau_i})_{1 \leq i \leq s}$. Regarding the last item, note that it covers those names that were introduced for unnamed elements in other canonical interpretations. For them, we only have to explicitly capture the non-determinism, w.r.t. basic concepts. All relations on the corresponding named individuals are completely determined by the assertions added in the item before (i.e., including those added in other iteration steps, maybe for some $\mathcal{I}_{\tau_j}$ with $j \neq i$). As mentioned above, we second have to ensure that the ABox created for a type also satisfies the conditions specified by it.

Definition 4 (Tree-ABox). *An ABox $\mathcal{A}$ produced by* TreeABox *given a name $b \in \mathsf{N}_\mathsf{I}$ as input is a* tree-ABox *for a type $\tau_i = (\mathcal{B}, \mathcal{M}, \mathcal{Q})$ if:*

- *$\langle \mathcal{O}, \mathcal{A} \cup \mathcal{A}_\mathcal{B}(b) \rangle$ is consistent;*
- *for all $\varphi \in \mathcal{Q}_\Phi$, $\varphi \in \mathcal{Q}$ iff $\langle \mathcal{O}, \mathcal{A} \cup \mathcal{A}_\mathcal{B}(b) \rangle \models \varphi$;*
- *for all $S \in \bigcup_{\varphi \in \mathcal{Q}_\Phi} 2^{\mathsf{N}_\mathsf{T}(\varphi)}$, $S \in \mathcal{M}$ iff there are a CQ $\varphi \in \mathcal{Q}_\Phi$ and a partial homomorphism $\pi\colon \mathsf{N}_\mathsf{T}(\varphi) \to \mathsf{N}_\mathsf{I}(\mathcal{A})$ of φ into $\mathcal{A}$ with $b \in \mathsf{range}(\pi)$, $S = \mathsf{dom}(\pi)$.*

Because of (*), the sizes of the tree-ABoxes are finite, and they are independent of the data: given $v := \max\{|\mathsf{N}_\mathsf{T}(\varphi)| \mid \varphi \in \mathcal{Q}_\Phi\}$, the maximal number of terms occurring in one of the CQs, and $t \leq 2^{|\mathbb{B}(\mathcal{O})|+|\mathcal{Q}_\Phi|+v*|\mathcal{Q}_\Phi|}$, the number of possible types, the depth of how far we specify the prototypical trees is bounded by $d := |\mathbb{B}(\mathcal{O})|^{t*|\mathsf{N}_\mathsf{R}^-(\mathcal{O})|^t*m} + m$; this follows from the facts that the names in $\mathsf{N}_\mathsf{I}^{\mathsf{tree}}$

[5] All names can be added, for example, by assuming $\top$ to be rigid.

are built from elements of $\mathsf{N}_\mathsf{R}^-(\mathcal{O})$, sometimes with subscripts from $[1,t]$, and that we consider these names in t different interpretations.

Finally, the additional data is a tuple as follows, polynomial in the data:

$$((\mathcal{A}'_i)_{0\leq i\leq n+k}, (\mathcal{T}_{a,i})_{\substack{a\in\mathsf{N}_\mathsf{I}(\mathcal{K})\cup\mathsf{N}_\mathsf{I}^{\mathsf{aux}},\\ 0\leq i\leq n+k}}, (\mathcal{A}_{\tau,i})_{\substack{\tau\in\mathbb{T},\\ 0\leq i\leq|\tau|}}),\text{ where}$$

- $\mathcal{A}'_i$ is an name-ABox for $\mathsf{N}_\mathsf{I}(\mathcal{K})\cup\mathsf{N}_\mathsf{I}^{\mathsf{aux}}$ for all $i\in[0,n+k]$, and all $\mathcal{A}'_i$ contain the same rigid assertions;
- $\mathcal{T}_{a,i}$ is a type for all $a\in\mathsf{N}_\mathsf{I}(\mathcal{K})\cup\mathsf{N}_\mathsf{I}^{\mathsf{aux}}$ and $i\in[0,n+k]$, and $\mathcal{T}_{a,i}^{(1)}=\mathsf{BT}(a,\mathcal{A}'_i)$.
- $(\mathcal{A}_{\tau_j})_{1\leq j\leq|\tau|}$ is the result of applying $\mathsf{TreeABox}$ to τ, some sequence of corresponding canonical interpretations, and a fresh name b for all $\tau\in\mathbb{T}$; and $\mathcal{A}_{\tau_i}$ is a tree-ABox for τ_i for all $i\in[1,|\tau|]$ if there is a name $a\in\mathsf{N}_\mathsf{I}(\mathcal{K})\cup\mathsf{N}_\mathsf{I}^{\mathsf{aux}}$ such that $\tau=\bigcup_{0\leq j\leq n+k}\{\mathcal{T}_{a,j}\}$.

For names $a,b\in\mathsf{N}_\mathsf{I}$ and a type τ_i, let $\mathcal{A}_{\tau_i}[b/a]$ be the ABox obtained from a tree-ABox $\mathcal{A}_{\tau_i}$ by replacing every b by a, also within individual names. For a tuple t as above, we define the ABox $\mathcal{A}^{\mathsf{t}}_{\mathsf{tree},i}$ for $i\in[0,n+k]$ as the set that contains, for all $a\in\mathsf{N}_\mathsf{I}(\mathcal{K})\cup\mathsf{N}_\mathsf{I}^{\mathsf{aux}}$, all assertions from $\mathcal{A}_{\tau_j}[b/a]$, where $\tau=\{\mathcal{T}_{a,i}\mid 0\leq i\leq n+k\}$ and $\tau_j=\mathcal{T}_{a,i}$. $\mathcal{A}^{\mathsf{t}}_{\mathsf{tree}}$ denotes the ABox that contains only the rigid assertions from $\mathcal{A}^{\mathsf{t}}_{\mathsf{tree},i}$, for all $i\in[0,n+k]$, and all names occurring in these ABoxes[6]; by construction, all of the latter ABoxes agree on those.

Lemma 1. *$\mathcal{W}$ is r-satisfiable w.r.t. ι and $\mathcal{K}$ iff there is a tuple*

$$\mathsf{t}=((\mathcal{A}'_i)_{0\leq i\leq n+k}, (\mathcal{T}_{a,i})_{\substack{a\in\mathsf{N}_\mathsf{I}(\mathcal{K})\cup\mathsf{N}_\mathsf{I}^{\mathsf{aux}},\\ 0\leq i\leq n+k}}, (\mathcal{A}_\tau)_{\tau\in\mathbb{T}})$$

as specified above such that, for all $i\in[0,n+k]$:

(C1) $\mathcal{K}^i_\mathsf{R}:=\langle\mathcal{O},\mathcal{A}'_i\cup\mathcal{A}_i\cup\mathcal{A}_{Q_{\iota(i)}}\cup\mathcal{A}^{\mathsf{t}}_{\mathsf{tree},i}\rangle$ *is consistent and,*
(C2) *for all $p_j\in\overline{W_{\iota(i)}}$, we have $\mathcal{K}^i_\mathsf{R}\not\models\varphi_j$.*

Proof. ($\Rightarrow$) As outlined in Section 2, we can consider the $n+k+1$ interpretations from the definition of r-satisfiability integrated within a single interpretation if we rename the flexible symbols accordingly. The advantage of this approach is that we can, w.l.o.g., assume that this interpretation is a canonical interpretation, which is not possible with the single interpretations from the definition of r-satisfiability because of the shared domain. Hence, for every $i\in[0,n+k+1]$ and every flexible name X in $\mathsf{N}_\mathsf{C}(\mathcal{O})\cup\mathsf{N}_\mathsf{R}(\mathcal{O})$, we introduce a fresh name $X^{(i)}$ called the *i-th copy* of X. If X is a more complex expressions (an axiom, CQ, or conjunction of CQ literals), $X^{(i)}$ is obtained by replacing every occurrence of a flexible name by its i-th copy. By [4, Lem. 4.14], $\mathcal{W}$ is r-satisfiable w.r.t. ι and $\mathcal{K}$ iff the conjunction $\chi_{\mathcal{W},\iota}$ of CQ literals has a model $\mathcal{I}$ w.r.t. $\langle\mathcal{O}_{\mathcal{W},\iota},\mathcal{A}\rangle$, where:

$$\chi_{\mathcal{W},\iota}:=\bigwedge_{1\leq i\leq k}\chi_i^{(n+1+i)}\wedge\bigwedge_{0\leq i\leq n}\chi_{\iota(i)}^{(i)},\quad \chi^{(i)}:=\bigwedge_{p_j\in W_{i-n-1}}\varphi_j^{(i)}\wedge\bigwedge_{p_j\in\overline{W}_{i-n-1}}\neg\varphi_j^{(i)},$$

$$\mathcal{O}_{\mathcal{W},\iota}:=\{\alpha^{(i)}\mid\alpha\in\mathcal{O},\ 0\leq i\leq n+k\},\quad \mathcal{A}:=\bigcup_{\substack{0\leq i\leq n,\\ \alpha\in\mathcal{A}_i}}\{\alpha^{(i)}\}.$$

[6] As above, this can be ensured, for example, by assuming $\top$ to be rigid.

For simplicity, we often focus on some $i \in [0, n+k]$ and consider the i-th copies of concept and role names as the (original) flexible names and disregard all other copies (but not the rigid names); in the following, we refer to those parts of $\mathcal{I}$, in which the signature is smaller and renamed, as $\mathcal{I}_i$. Similarly, we consider $\mathsf{T}(a, \mathcal{I}_i)$ to be the *type* of a in $\mathcal{I}$ for $i \in [0, n+k]$, and may refer to the set of all these types as the *temporal type* of a in $\mathcal{I}$.

We then can define the components of the required tuple t for $i \in [0, n+k]$ and $c \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$ easily: $\mathcal{T}_{c,i} := \mathsf{T}(c, \mathcal{I}_i)$ and

$$\mathcal{A}'_i := \{(\neg)B(a) \mid a \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}, B \in \mathbb{B}(\mathcal{O}), a \in B^{\mathcal{I}_i}(a \notin B^{\mathcal{I}_i})\} \cup \\ \{(\neg)R(a, b) \mid a, b \in \mathsf{N_I}(\mathcal{K}), R \in \mathsf{N_R}(\mathcal{O}), (a, b) \in R^{\mathcal{I}_i}((a, b) \notin R^{\mathcal{I}_i})\}.$$

For all $\tau \in \mathbb{T}$, we choose an arbitrary $a \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$ with temporal type τ in $\mathcal{I}$; for each $i \in [1, |\tau|]$, define $\mathcal{I}_{\tau_i}$ to be some $\mathcal{I}_{j|a}$ with $\tau_i = \mathsf{T}(a, \mathcal{I}_j)$; and then define $(\mathcal{A}_{\tau_i})_{1 \leq i \leq |\tau|} := \mathsf{TreeABox}(\tau, (\mathcal{I}_{\tau_i})_{1 \leq i \leq |\tau|}, a)$; if there is no such element a, the ABoxes are empty. We can assume the latter algorithm to iterate only once over the types because the interpretations of the new names in all the interpretations are given already, by the interpretations of the elements for which they were introduced, respectively (i.e., we do not have to extend interpretations).

It is easy to see that the tuple is as required—that is, $\mathcal{I}_i$ represents the model of $\mathcal{K}^i_{\mathsf{R}}$—if $(*)$ we assume $\mathcal{I}$ to be such that (1) the trees of unnamed successors for two names $a, b \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$ that have the same temporal type in $\mathcal{I}$ are isomorphic w.r.t. the rigid symbols and that (2) it interprets those successors the same in $\mathcal{I}_i$ and $\mathcal{I}_j$ if $\mathsf{T}(a, \mathcal{I}_i) = \mathsf{T}(a, \mathcal{I}_j)$. This is the case because $\mathcal{A}^{\mathsf{t}}_{\mathsf{tree},i}$ then is trivially satisfied for all $i \in [0, n+k]$: the interpretations that can be selected for the construction of a prototypical ABox $\mathcal{A}_\tau$ then all are isomorphic; that is, the construction neither depends on the name chosen as prototype for the temporal type nor on the indexes i whose interpretation $\mathcal{I}_i$ is chosen as prototype for a type. We lastly show that $(*)$ is a valid assumption. The proof is by contradiction. We hence assume that such a given model $\mathcal{I}$ cannot be simplified in the described way without loosing the property that it satisfies both the KB and the conjunction of CQ literals. Let $\mathcal{J}$ be a corresponding adaptation of such a model $\mathcal{I}$, constructed as follows:

– For every temporal type τ for which there is an individual name a such that τ is the temporal type of a in $\mathcal{I}$, and for every $j \in [1, |\tau|]$, select one index $i_j \in [0, n+k]$ such that $\tau_j = \mathsf{T}(a, \mathcal{I}_{i_j})$, and let $\mathcal{A}_{\tau,j}$ be the (possibly infinite) set of assertions representing $\mathcal{I}_{i_j}$ on all unnamed successors of a. That is, $\mathcal{A}_{\tau,j}$ covers all the successors contained in $\mathcal{I}$, but it only describes $\mathcal{I}$ w.r.t. the rigid names and the i_j-th copies of flexible names.
– Adapt $\mathcal{I}$ as follows. For every $a \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$ with temporal type τ in $\mathcal{I}$, replace all unnamed successors by copies of the elements occurring in $\mathcal{A}_{\tau,j}$ (for an arbitrary $j \in [1, |\tau|]$). For every $i \in [0, n+k]$, the interpretation of the i-th copies of names on these elements in $\mathcal{J}$ is then given by the corresponding flexible assertions in the one set $\mathcal{A}_{\tau,j}$ for which we have $\tau_j = \mathsf{T}(a, \mathcal{I}_i)$. The interpretation of the rigid names is also given by these ABoxes since they all agree on those

names. As with $\mathcal{I}$, we use interpretations $\mathcal{J}_i$ to refer to $\mathcal{J}$ on the rigid names and on the i-th copies; that is, we consider the i-th copies as the (original) flexible names and disregard all other copies.

– The construction of $\mathcal{J}$ maintains the types, meaning that $\mathsf{T}(a, \mathcal{J}_i) = \mathsf{T}(a, \mathcal{I}_i)$ for all $a \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$, $i \in [0, n+k]$. Let $(\mathcal{B}, \mathcal{M}, \mathcal{Q}) := \mathsf{T}(a, \mathcal{I}_i)$, $(\mathcal{B}', \mathcal{M}', \mathcal{Q}') := \mathsf{T}(a, \mathcal{J}_i)$. Note that, to replace the unnamed successors of a in $\mathcal{I}$, we chose unnamed successors of some $b \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$ of the same temporal type as a in $\mathcal{I}$; and that the interpretation in $\mathcal{J}_i$ on a and its unnamed successors is given by the one on b and its unnamed successors in some $\mathcal{I}_j$ with $\mathsf{T}(b, \mathcal{I}_j) = \mathsf{T}(a, \mathcal{I}_i)$. Clearly, for every homomorphism of some $\varphi \in \mathcal{Q}'$ into $\mathcal{J}_{i|a}$, there is a corresponding one into $\mathcal{I}_{j|b}$ if a does not occur in φ, by the definition of $\mathcal{J}_i$. This especially holds because the interpretation of the rigid symbols on a and its unnamed successors in the whole interpretation $\mathcal{J}$ is fully determined by the interpretation of the rigid symbols in $\mathcal{I}_{j|b}$. If a occurs in φ, then the definition of $\mathsf{T}(a, \mathcal{I}_i)$ yields $\{a\} \in \mathcal{M}$, and $\mathsf{T}(b, \mathcal{I}_j) = \mathsf{T}(a, \mathcal{I}_i)$ hence implies $a = b$ by the definition of $\mathsf{T}(b, \mathcal{I}_j)$. The construction of $\mathcal{J}_{i|a}$ then, as before, yields $\mathcal{Q}' \subseteq \mathcal{Q}$. For the other direction, we consider a homomorphism of some $\varphi \in \mathcal{Q}$ into $\mathcal{I}_i$. But then we also have one of φ into $\mathcal{I}_j$ which, in turn, yields a corresponding one into $\mathcal{J}_i$. $\mathcal{M} = \mathcal{M}'$ follows by analogous arguments. We show $\mathcal{B} = \mathcal{B}'$. Since we do not adapt the interpretation of the concept and role names w.r.t. only named individuals, it is left to focus on basic concepts of the form $\exists R$. The only critical case is thus the one where there is an element u_{aR} in $\mathcal{I}$ but not in $\mathcal{J}$ and a has no named R-successor (in both $\mathcal{I}$ and $\mathcal{J}$) either; that is, there is no element u_{bR} in $\mathcal{I}$, but b has a named R-successor (again in both $\mathcal{I}$ and $\mathcal{J}$) by $\mathsf{BT}(a, \mathcal{I}_i) = \mathsf{BT}(b, \mathcal{I}_j)$. However, note that we assume the ontology to contain all CIs of the form $\exists R \sqsubseteq \exists R$. Hence, the element u_{bR} must exist in $\mathcal{I}$ by Definition 1. By construction, we thus get that u_{aR} exists in $\mathcal{J}$, which contradicts the assumption. The case where an element u_{aR} exists in $\mathcal{J}$ but not in $\mathcal{I}$ is not critical w.r.t. possible changes of the basic type.

Since we do not adapt the interpretation of the concept and role names w.r.t. only named elements, $\mathcal{J} \models \mathcal{A}$. Regarding the named elements, $\mathcal{J}$ also satisfies $\mathcal{O}_{\mathcal{W},\iota}$ since the adaptation retains the basic types. Regarding an unnamed element e, observe that it is valid to argument based on the single interpretations $\mathcal{J}_i$, $i \in [0, n+k]$, instead of on $\mathcal{J}$ as a whole because the ontology contains no axioms where different kinds of copies occur in and $\mathcal{J}_i$ represents the interpretation of rigid names in $\mathcal{J}$. The interpretation of e in such a $\mathcal{J}_i$ corresponds to the interpretation of an isomorphic element in some $\mathcal{I}_j$, $j \in [0, n+k]$. But we have $\mathcal{I} \models \mathcal{O}_{\mathcal{W},\iota}$, and $\mathcal{O}_{\mathcal{W},\iota}$ contains all copies (i.e., also the j-th) of all axioms in $\mathcal{O}$. So $\mathcal{J}_i$ satisfies all axioms in $\mathcal{O}$, which yields $\mathcal{J} \models \mathcal{O}_{\mathcal{W},\iota}$. $\mathcal{J}$ also satisfies all CQ literals $\varphi^{(i)}$ satisfied in $\mathcal{I}$. This is clear if only named individuals are considered because $\mathcal{I}$ and $\mathcal{J}$ agree on the interpretation of concept and role names w.r.t. only named individuals and the basic types are maintained. For the other cases, observe that we can argument based on single interpretations $\mathcal{I}_i/\mathcal{J}_i$, again, since the copies of the CQs only contain one kind of copies of names, and all these interpretations for $\mathcal{I}/\mathcal{J}$ agree on the rigid symbols. We consider the case where there is a homomorphism that maps to maximally one named individual. All the unnamed elements in the range must form a tree structure with a named individual $a \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$ as root. Assuming $\mathsf{T}(a, \mathcal{I}_i) = (\mathcal{B}, \mathcal{M}, \mathcal{Q})$, we must have $\varphi \in \mathcal{Q}$ by Definition 3. From $\mathsf{T}(a, \mathcal{I}_i) = \mathsf{T}(a, \mathcal{J}_i)$, we get that $\varphi^{(i)}$ also in $\mathcal{J}$ is satisfied based on a and its unnamed successors. If there is a homomorphism π mapping to both several named and unnamed elements, then a corresponding homomorphism π' into $\mathcal{J}$ can be obtained based on the type. By Definition 3, for all $a \in (\mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}) \cap \mathsf{range}(\pi)$, there is a set of terms V_a in $\mathsf{T}(a, \mathcal{I}_i)^{(2)}$ containing exactly the terms $t \in \mathsf{N_T}(\varphi)$ with $\pi(t) = a$ or $\pi(t)$ being an unnamed successor of a. From $\mathsf{T}(a, \mathcal{J}_i) = \mathsf{T}(a, \mathcal{I}_i)$, we get $V_a \in \mathsf{T}(a, \mathcal{J}_i)^{(2)}$, which means that there must be a corresponding partial homomorphism π_a of φ into $\mathcal{J}_i$, again by Definition 3. Now, we define π': (i) $\pi'(t) := \pi(t)$ if $\pi(t) \in \mathsf{N_I}(\mathcal{K} \cup \mathsf{N_I^{aux}})$. (ii) $\pi'(t) := \pi_a(t)$ for all $a \in (\mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}) \cap \mathsf{range}(\pi)$ and $t \in V_a$. Regarding the negative CQ literals, we proceed by contradiction and assume that $\mathcal{J} \models \varphi_j^{(i)}$ for some $p_j \in \overline{W_{\iota(i)}}$. Again, we can consider single interpretations $\mathcal{I}_i/\mathcal{J}_i$. Since we do not adapt the interpretation of the names on only named elements the corresponding homomorphism π must map to unnamed elements. It can neither map to only unnamed elements and maximally to one named element: the arguments correspond to those given above for the corresponding case for the positive CQ literals if $\mathcal{I}_i$ and $\mathcal{J}_i$ and $\mathcal{I}$ and $\mathcal{J}$ are switched, respectively. The same holds for the case where π maps to several named and to unnamed elements.

($\Leftarrow$) We regard a tuple t that satisfies the conditions from Lemma 1 and construct interpretations $\mathcal{I}_0, \ldots, \mathcal{I}_n$, $\mathcal{J}_1, \ldots, \mathcal{J}_k$ as required based on the KBs in Condition (C1), which contain all the necessary information (i.e., $\mathcal{K}_\mathsf{R}^i$ yields $\mathcal{I}_i$ for $i \in [0, n]$ and $\mathcal{J}_i$ for $i \in [n+1, n+k]$): we have the name-ABoxes for the named individuals in the TKB, and $\mathcal{A}^\mathsf{t}_{\mathsf{tree},i}$ for all $i \in [0, n+k]$ for those successors (up to some "depth" $d' \leq d$) of the latter individuals that must exist given $\mathcal{O}$. The shared domain of our interpretations contains all of the names occurring in these KBs and additional unnamed elements, to include the required successors of depths greater than d'. The idea is to inductively introduce elements by continuing the repetition we have given the construction of the tree-ABoxes.

The ontology and the ABoxes $\mathcal{A}_i$ for $i \in [0, n+k]$ are clearly satisfied by the respective interpretations, since the KBs $\mathcal{K}_\mathsf{R}^i$ are consistent and our interpretations are completely defined based on canonical ones (i.e., the ones used to construct the ABoxes $\mathcal{A}^\mathsf{t}_{\mathsf{tree},i}$; these ABoxes describe the interpretations completely in the sense that they capture all the necessary rigid knowledge and nondeterministic decisions). The additional ABoxes $\mathcal{A}_{\mathcal{Q}_{\iota(i)}}$ ensure that the positive CQ literals are satisfied as required. By contradiction, it can easily be shown that no negative CQ literal is satisfied. By (C2), a corresponding homomorphism π must map to unnamed elements and, since we untangle the interpretations during the construction of $\mathcal{A}^\mathsf{t}_{\mathsf{tree},i}$ to depth $\geq 2m$, it cannot map to elements in $\mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$. This yields a contradiction to (C2) because there are isomorphic named elements for all those π maps to, since the repeating trees are of depth m. □

Regarding complexity: $\mathcal{W}$ and ι can be guessed in polynomial time, and the LTL satisfiability testing w.r.t. a given $\mathcal{W}$ and ι can be done in polynomial time

[4, Lem. 4.12]. Similarly, the tuple from Lemma 1 can be guessed in polynomial time; note that the tests if the ABoxes for the temporal types are tree-ABoxes are data independent. Since both KB consistency and CQ non-entailment are in NP in $\textit{DL-Lite}^{\mathcal{H}}_{krom}$ [1, Thm. 8.2], (C1) and (C2) can be decided nondeterministically in polynomial time. TCQ satisfiability is thus in NP [4, Lem. 4.7].

Theorem 1. *TCQ entailment regarding a TKB in* $\textit{DL-Lite}^{\mathcal{H}}_{krom}$ *is in* co-NP.

The expressivity of TCQs allows to reduce TCQ entailment in much more expressive DLs to TCQ entailment in $\textit{DL-Lite}^{\mathcal{H}}_{krom}$ [6], without an impact on the data. For example, CIs[7] as $\exists R.A_1 \sqsubseteq A_2$ and $A_1 \sqsubseteq A_2 \sqcup A_3$ can be encoded using TCQs $\neg\exists xy.R(x,y) \wedge A_1(y) \wedge \overline{A_2}(x)$ and $\neg\exists x.A_1(x) \wedge \overline{A_2}(x) \wedge \overline{A_3}(x)$ if the ontology is extended by the CIs $\neg A_2 \sqsubseteq \overline{A_2}$ and $\neg A_3 \sqsubseteq \overline{A_3}$. This yields:

Corollary 1. *TCQ entailment regarding a TKB in* $\mathcal{ALCHI}$ *is in* co-NP.

4 Conclusions

We have shown that the data complexity of TCQ entailment w.r.t. temporal knowledge bases in expressive DLs is in co-NP, even if rigid symbols are considered. This result is interesting since already standard conjunctive query entailment is co-NP-hard, which means we get the temporal features "for free". Yet, it remains to design deterministic algorithms to translate the result into practice.

References

1. Artale, A., Calvanese, D., Kontchakov, R., Zakharyaschev, M.: The *DL-Lite* family and relations. Journal of Artificial Intelligence Research 36, 1–69 (2009)
2. Baader, F., Borgwardt, S., Lippmann, M.: Temporalizing ontology-based data access. In: Proc. of CADE. pp. 330–344 (2013)
3. Baader, F., Borgwardt, S., Lippmann, M.: Temporal conjunctive queries in expressive description logics with transitive roles. In: Proc. of AI. pp. 21–33 (2015)
4. Baader, F., Borgwardt, S., Lippmann, M.: Temporal query entailment in the description logic $\mathcal{SHQ}$. Journal of Web Semantics 33, 71–93 (2015)
5. 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, 2 edn. (2007)
6. Borgwardt, S., Thost, V.: Temporal query answering in *DL-Lite* with negation. In: Proc. of GCAI. pp. 51–65 (2015)
7. Borgwardt, S., Thost, V.: Temporal query answering in the description logic $\mathcal{EL}$. In: Proc. of IJCAI. pp. 2819–2825 (2015)
8. Deutsch, A., Nash, A., Remmel, J.B.: The chase revisited. In: Proc. of PODS. pp. 149–158 (2008)
9. Pnueli, A.: The temporal logic of programs. In: Proc. of FOCS. pp. 46–57 (1977)
10. Poggi, A., Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Linking data to ontologies. Journal on Data Semantics 10, 133–173 (2008)
11. Thost, V.: Using Ontology-Based Data Access to Enable Context Recognition in the Presence of Incomplete Information. Ph.D. thesis, TU Dresden (2017)

[7] The syntax and semantics of many DLs extending $\textit{DL-Lite}^{\mathcal{H}}_{krom}$ are described in [5].