Complexity of Temporal Query Abduction in DL-*Lite*

Szymon Klarman and Thomas Meyer

Centre for Artificial Intelligence Research,
CSIR Meraka and University of KwaZulu-Natal, South Africa
{sklarman,tmeyer}@csir.co.za

Abstract. Temporal query abduction is the problem of hypothesizing a minimal set of temporal data which, given some fixed background knowledge, warrants the entailment of the query. This problem formally underlies a variety of forms of explanatory and diagnostic reasoning in the context of time series data, data streams, or otherwise temporally annotated structured information. In this paper, we consider (temporally ordered) data represented in Description Logics from the popular DL-*Lite* family and Temporal Query Language, based on the combination of LTL with conjunctive queries. In this defined setting, we study the complexity of temporal query abduction, assuming different restrictions on the problem and minimality criteria for abductive solutions. As a result, we draw several revealing demarcation lines between NP-, DP- and PSpace-complete variants of the problem.

1 Introduction

The ubiquity and importance of time-related information in semantic applications necessitates the need for novel representation and reasoning solutions for dealing with the temporal dimension of data. In particular, the task of querying temporal data in the presence of ontological constraints has in recent years gained considerable attention within the Semantic Web and Description Logic communities [20,4,12,2,13]. Query languages supporting complex temporal patterns are essential for enabling fine-grained retrieval and analysis of temporally annotated information, both in the static settings, such as involving historical or time series data, as well as in the dynamic ones, witnessed in the context of streaming data applications. One interesting derived problem which we study in this paper is *temporal query abduction*, i.e., the problem of hypothesizing a minimal set of temporal data which, given some fixed background knowledge, warrants the entailment of the query. This type of inference is inherent to a variety of explanatory and diagnostic forms of reasoning applicable in the context of temporal data. For instance, suppose that the fact that the grass is frozen in some location x follows whenever the following query is satisfied:

$$[\mathit{rainIn}(x)]\ \mathsf{X}\ [\exists y.(\mathit{tempIn}(y,x) \wedge \mathit{below\_0}(y))]$$

meaning that the below-zero temperature occurred in location x immediately after a rainfall. Using abductive inference, one is then able to explain this fact with a hypothesis $\mathcal{A}_{i+1} = \{(\mathit{tempIn}(t, x), \mathit{below\_0}(t)\}$, provided he already knows that $\mathcal{A}_i = \{\mathit{rainIn}(x)\}$, where $\mathcal{A}_i$ and $\mathcal{A}_{i+1}$ represent the data sets that hold in some relative time points i and $i + 1$.

Following the popular paradigm of ontology-based data access, we consider data expressed as Description Logic (DL) axioms, accessed through an ontological layer expressed in logics from the popular DL-*Lite* family [6]. Further, we use Temporal Query Language (TQL), originally introduced in [13], for querying temporally ordered DL data. TQL, based on the combination of Linear Temporal Logic with conjunctive queries, offers a flexible means for interleaving data patterns with temporal constraints, thus providing a powerful framework for expressing interesting correlations in temporal data sets. Based on this foundation, we define the problem of temporal query abduction, which is the central conceptual contribution of this work. On the technical side, we analyse the computational complexity of the problem, considering different fragments of the temporal language and different restrictions on the space of abductive solutions. As a result, we draw several revealing demarcation lines between NP-, DP- and PSpace-complete variants of the problem.

A number of other types of abductive inference for DLs have been studied in the literature. These include: concept abduction [8,3], TBox and ABox abduction [9,14,15,18] and query/rule abduction [10,7,21]. This last form of reasoning, dealing with the identification of minimal ABoxes satisfying a certain query/rule, is most closely related to the problem studied in this paper, although it does not consider the temporal dimension of DL data. To the best of our knowledge, the problem of temporal query abduction has never been formulated within the DL framework before, even though it has been properly recognized and investigated on the grounds of other formalisms [22,5].

The paper is organized as follows. In the next section we recap preliminaries of DLs and Temporal Query Language. In Section 3 we introduce the problem of temporal query abduction. Then, in Section 4 we present the complexity results and discuss their consequences on the main problem. The paper is concluded in Section 5. The proofs of the results are included in the online technical report [17].

The results reported in this paper are minor generalizations of those originally presented by the authors in [16].

2 Preliminary Notions

In this section, we introduce the basic nomenclature regarding Description Logics, conjunctive queries, and the representation and querying of temporal data.

2.1 DL-*Lite* and Conjunctive Queries

A *Description Logic* (DL) language is given by a vocabulary $\Sigma = (\mathsf{N_I}, \mathsf{N_C}, \mathsf{N_R})$ and a set of logical constructors [1]. The vocabulary consists of countably infinite sets of individual names ($\mathsf{N_I}$), concept names ($\mathsf{N_C}$) and role names ($\mathsf{N_R}$). An ABox $\mathcal{A}$ is a finite set of assertions $A(a)$ and $r(a, b)$, for $a, b \in \mathsf{N_I}$, $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. A TBox $\mathcal{T}$ is a finite set of terminological axioms, e.g., concept and role inclusions, whose precise syntax is determined by the given DL. The semantics is given in terms of DL interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, defined as usual [1]. An interpretation $\mathcal{I}$ is a model of $\mathcal{T}$ and $\mathcal{A}$, denoted as $\mathcal{I} \models \mathcal{T}, \mathcal{A}$, *iff* it satisfies every axiom in $\mathcal{T}$ and $\mathcal{A}$. If $\mathcal{T}$ and $\mathcal{A}$ have a common model they are said to be consistent.

Abiding by the nomenclature of ontology-based data access paradigm, we consider the ABox as *data* and the TBox as the *ontology*, which provides an additional semantic layer over the data, thus enriching the querying capabilities. A *conjunctive query* (CQ) over a DL vocabulary Σ is a first-order formula $\exists \boldsymbol{y}.\varphi(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x}, \boldsymbol{y}$ are sequences of variables, from a countably infinite set of variables $\mathsf{N_V}$. The sequence $\boldsymbol{x}$ denotes the free (answer) variables in the query, while $\boldsymbol{y}$ the quantified ones. The formula φ is a conjunction of atoms over $\mathsf{N_C}, \mathsf{N_R}$ of the form $A(u), r(u, v)$, where $u, v \in \mathsf{N_V} \cup \mathsf{N_I}$ are called terms. By $\mathsf{term}(q)$ we denote the set of all terms occurring in a CQ q and by $\mathsf{avar}(q)$ the set of all its answer variables. We call q boolean whenever $\mathsf{avar}(q) = \emptyset$. A boolean CQ q is satisfied in $\mathcal{I}$ *iff* there exists a mapping $\mu : \mathsf{term}(q) \mapsto \Delta^{\mathcal{I}}$, with $\mu(a) = a^{\mathcal{I}}$ for every $a \in \mathsf{N_I}$, such that for every $A(u)$ and $r(u, v)$ in q it is the case that $\mu(u) \in A^{\mathcal{I}}$ and $(\mu(u), \mu(v)) \in r^{\mathcal{I}}$. We say that q is entailed by a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$, denoted as $\mathcal{T}, \mathcal{A} \models q$ *iff* q is satisfied in every model of $\mathcal{T}$ and $\mathcal{A}$. An *answer* to q is a mapping σ such that $\sigma : \mathsf{avar}(q) \mapsto \mathsf{N_I}$. By $\sigma(q)$ we denote the result of uniformly substituting every occurrence of x in q with $\sigma(x)$, for every $x \in \mathsf{avar}(q)$. An answer σ is called *certain* over $\mathcal{T}, \mathcal{A}$ *iff* $\mathcal{T}, \mathcal{A} \models \sigma(q)$. The set of all certain answers to q over $\mathcal{T}, \mathcal{A}$ is denoted by $\mathsf{cert}(q, \mathcal{T}, \mathcal{A})$. By $\mathcal{Q}_{\Sigma}$ we denote the class of all conjunctive queries over the vocabulary Σ.

In this paper, we focus on logics from the DL-*Lite* family [6], such as DL-*Lite*$_{\mathcal{R}}$, DL-*Lite*$_{\mathcal{F}}$ or DL-*Lite*$_{\mathcal{A}}$, underlying the OWL 2 QL ontology language profile[1], for which CQs enjoy the so-called first-order rewritability property, defined as follows.

Definition 1 (FO rewritability [6]). *For every CQ $q \in \mathcal{Q}_{\Sigma}$ and a TBox $\mathcal{T}$, there exists a FO formula $q^{\mathcal{T}}$ such that for every ABox $\mathcal{A}$ and answer σ to q, it holds that $\sigma \in \mathsf{cert}(q, \mathcal{T}, \mathcal{A})$* iff *$db(\mathcal{A}) \Vdash \sigma(q^{\mathcal{T}})$, where $db(\mathcal{A})$ denotes $\mathcal{A}$ considered as a database/FO interpretation and $\Vdash$ is the FO satisfaction relation.*

Recall, that given $\mathcal{T}$ in any of such DLs and a boolean q, the FO rewriting $q^{\mathcal{T}}$ of q is a union of possibly exponentially many CQs, including q. The number of these CQs is bounded by $\ell(\mathcal{T})^{\ell(q)}$, where $\ell(\dagger)$ denotes the size of the input $\dagger$ measured in the total number of symbols used. Every CQ q' in $q^{\mathcal{T}}$ is such that $\mathcal{T} \cup \{q'\} \models q$ and its size is linear in $\ell(q)$. The query entailment problem is NP-complete in the combined complexity, even when the TBox is empty, while checking consistency of $\mathcal{T}, \mathcal{A}$ is in PTime [6].

[1] See `http://www.w3.org/TR/owl2-profiles/`.

Regardless of this default focus, many of the results presented here can be naturally extended to other DLs with similar characteristics, such as other members of the DL-*Lite* family or logics in the $\mathcal{EL}$ family [19].

2.2 Temporal Query Language

We consider a discrete, linear *time domain* $(\mathbb{Z}, <)$, with integers representing *time points* ordered by the smaller-than relation. An *interval* over $\mathbb{Z}$ is a set $I = [I^-, I^+] = \{i \in \mathbb{Z} \mid I^- \leq i \leq I^+\}$, where $I^- \leq I^+ \in \mathbb{Z} \cup \{-\infty, +\infty\}$ and $-\infty < i < +\infty$ for every $i \in \mathbb{Z}$.

Definition 2 (A-sequence). *An* A-sequence $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ *is a sequence of ABoxes, for some interval I over $\mathbb{Z}$.*

A-sequences represent collections of datasets ordered w.r.t. the underlying time domain. The ordering of the ABoxes follows the smaller-than ordering of their indices. An A-sequence $\mathfrak{A}$ is said to be consistent with a TBox $\mathcal{T}$ if every ABox in it is consistent with $\mathcal{T}$. Consider A-sequences $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ and $\mathfrak{B} = (\mathcal{B}_i)_{i \in J}$. We use the following notation:

- $\mathcal{T}, \mathfrak{A} \models \mathfrak{B}$ ($\mathfrak{A} \models \mathfrak{B}$) *iff* $J \subseteq I$ and $\mathcal{T}, \mathcal{A}_i \models \mathcal{B}_i$ ($\mathcal{A}_i \models \mathcal{B}_i$) for every $i \in J$,
- $\mathfrak{A} \uplus \mathfrak{B}$, whenever $I \cap J \neq \emptyset$, to denote the A-sequence $(\mathcal{C}_i)_{i \in I \cup J}$ such that:
  - $\mathcal{C}_i = \mathcal{A}_i$, for every $i \in I \setminus J$,
  - $\mathcal{C}_i = \mathcal{B}_i$, for every $i \in J \setminus I$,
  - $\mathcal{C}_i = \mathcal{A}_i \cup \mathcal{B}_i$, for every $i \in I \cap J$,
- $\mathfrak{A} \rightharpoonup_n \mathfrak{B}$, for some $n \in I \cap J$, *iff* there exists a mapping $f : I \mapsto J$, such that:
  - $f(n) = n$,
  - $i < j$ *iff* $f(i) < f(j)$, for every $i, j \in I$,
  - $\mathcal{A}_i = \mathcal{B}_{f(i)}$, for every $i \in I$,

Next, we recall a variant of Temporal Query Language, proposed in [13], which we use for accessing A-sequences. It is a lightweight combination of Linear Temporal Logic (LTL) [23] with CQs, where CQs are embedded in the temporal language using the epistemic semantics.

Definition 3 (Temporal Query Language). *The* temporal query language *(TQL) over a class of conjunctive queries $\mathcal{Q}_\Sigma$ is the smallest set of formulas induced by the grammar:*

$$\phi, \psi \ ::= [q] \ \mid \ \neg\phi \ \mid \ \phi \wedge \psi \ \mid \ \phi \mathsf{U} \psi \ \mid \ \phi \mathsf{S} \psi$$

where $q \in \mathcal{Q}_\Sigma$. By $\mathsf{avar}(\phi)$ *we denote the set of free variables in ϕ. A TQL formula ϕ is called* boolean *whenever* $\mathsf{avar}(\phi) = \emptyset$*. The entailment relation for boolean TQL formulas w.r.t. an A-sequence $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ under a TBox $\mathcal{T}$ in time $i \in I$ is defined inductively as follows:*

$$\begin{array}{lll}
\mathcal{T}, \mathfrak{A}, i \models [q] & \textit{iff} & \mathcal{T}, \mathcal{A}_i \models q, \\
\mathcal{T}, \mathfrak{A}, i \models \neg\phi & \textit{iff} & \mathcal{T}, \mathfrak{A}, i \not\models \phi, \\
\mathcal{T}, \mathfrak{A}, i \models \phi \wedge \psi & \textit{iff} & \mathcal{T}, \mathfrak{A}, i \models \phi \textit{ and } \mathcal{T}, \mathfrak{A}, i \models \psi, \\
\mathcal{T}, \mathfrak{A}, i \models \phi \mathsf{U} \psi & \textit{iff} & \textit{there exists } j \in I \textit{ with } j > i \textit{ such that} \\
 & & \mathcal{T}, \mathfrak{A}, j \models \psi \textit{ and } \mathcal{T}, \mathfrak{A}, k \models \phi \textit{ for every } k \in I \\
 & & \textit{with } i < k < j, \\
\mathcal{T}, \mathfrak{A}, i \models \phi \mathsf{S} \psi & \textit{iff} & \textit{there exists } j \in I \textit{ with } j < i \textit{ such that} \\
 & & \mathcal{T}, \mathfrak{A}, j \models \psi \textit{ and } \mathcal{T}, \mathfrak{A}, k \models \phi \textit{ for every } k \in I \\
 & & \textit{with } i > k > j.
\end{array}$$

An answer *to a TQL formula ϕ is a mapping $\sigma : \mathsf{avar}(\phi) \mapsto \mathsf{N_I}$. By $\sigma(\phi)$ we denote the result of uniformly substituting every occurrence of x in ϕ with $\sigma(x)$, for every $x \in \mathsf{avar}(\phi)$. An answer σ is called* certain *over $\mathcal{T}, \mathfrak{A}$ at $i \in I$* iff *$\mathcal{T}, \mathfrak{A}, i \models \sigma(\phi)$. The set of all such answers is denoted by $\mathsf{cert}_i(\phi, \mathcal{T}, \mathfrak{A})$.*

As usual, using the operators U (strict *until*) and S (strict *since*), we can easily define other ones, such us $\mathsf{F}\phi = \top\mathsf{U}\phi$ (*some time in future*), $\mathsf{P}\phi = \top\mathsf{S}\phi$ (*some time in past*), $\mathsf{X}\phi = \bot\mathsf{U}\phi$ (*next in future*), $\mathsf{X}^{-}\phi = \bot\mathsf{S}\phi$ (*next in past*). In fact, LTL with U and S, which captures precisely the temporal component of TQL, is known to be expressively complete over $(\mathbb{Z}, <)$ [11]. Apart from the full TQL, in what follows we consider also some of its strict subsets. By $\text{TQL}^{\mathsf{F},\mathsf{P}}$ we denote the fragment where U and S appear only in the forms allowed in the definitions of F and P. Further, with TQL^{+} we refer to the positive fragment of TQL, i.e., TQL without the negation operator. Finally, by $\text{TQL}^{\mathsf{F},\mathsf{P},+}$, we denote the intersection of $\text{TQL}^{\mathsf{F},\mathsf{P}}$ and TQL^{+}.

Observe that given the epistemic interpretation of the embedded CQs, $[q]$ reads as "*q is entailed in the given time instant*", for a boolean CQ q. We can immediately paraphrase this interpretation by invoking the FO rewriting of q, in the sense of Definition 1. Note that the following correspondences immediately hold:

$$\mathcal{T}, \mathfrak{A}, i \models [q] \quad \textit{iff} \quad \mathcal{T}, \mathcal{A}_i \models q \quad \textit{iff} \quad db(\mathcal{A}_i) \Vdash q^{\mathcal{T}}.$$

Consequently, the negation $\neg[q]$ is naturally interpreted as negation-as-failure, reading "*it is not true that q is entailed in the given time instant*". This warrants the following equivalences:

$$\mathcal{T}, \mathfrak{A}, i \models \neg[q] \quad \textit{iff} \quad \mathcal{T}, \mathcal{A}_i \not\models q \quad \textit{iff} \quad db(\mathcal{A}_i) \not\Vdash q^{\mathcal{T}}.$$

These observations are critical for the work presented in this paper, as they allow to study satisfaction of TQL formulas by decoupling the temporal component of the problem from the CQ component, and addressing the latter, without loss of correctness, by applying the standard FO rewriting techniques and results, recalled in Section 2.1. Importantly, such lightweight combination of languages allows also for a modular reuse of existing temporal reasoners and highly optimized, efficient query answering engines [13]. In TQL^{+} the epistemic interpretation of CQs becomes redundant, and the resulting language coincides with the one proposed in [4].

3 Temporal Query Abduction

Temporal queries can be naturally used for formalizing temporal data patterns and correlations between them, which are known to occur in a given domain. For instance, consider the TQL formulas ϕ and ψ:

$$\phi = \\ \neg[\exists y.(\mathit{tempIn}(y,x) \land \mathit{above\_0}(y))] \; \mathsf{S} \; ([\exists y.(\mathit{tempIn}(y,x) \land \mathit{below\_0}(y))] \land \mathsf{X}^{-}[\mathit{rainIn}(x)])$$

$$\psi = \exists y.(\mathit{Grass}(y) \land \mathit{locIn}(y,x) \land \mathit{Frozen}(y))$$

The first one describes a location x, which experienced a rainfall in the past, followed directly by a below-zero temperature, after which no above-zero temperature has been recorded. The second one states that grass is frozen in a certain location x. Suppose there actually exists a causal dependency reflected via the rule $\phi \to \psi$, so that for any answer $\sigma = \{x \mapsto l\}$, whenever $\mathcal{T}, \mathfrak{A}, n \models \sigma(\phi)$ then $\mathcal{T}, \mathfrak{A}, n \models \sigma(\psi)$. In common diagnostic scenarios, such rules can be further employed to guide the explanation-finding for the observed data: whenever $\sigma(\psi)$ is true one might hypothesize a suitable collection of temporal data which makes $\sigma(\phi)$ true. This latter form of hypothetical inference, from a temporal query to temporal data, is a variant of classical abductive reasoning, which we study here and formalize it using the nomenclature coined in [9,15,7].

Definition 4 (Temporal query abduction). *A temporal query abduction (TQA) problem is a tuple $\Omega = (\mathcal{T}, \mathfrak{A}, \phi, n)$, where $\mathcal{T}$ is a TBox, $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ is an A-sequence, for some finite I, ϕ is a boolean TQL formula, and $n \in I$. A* solution *to Ω is an A-sequence $\mathfrak{D} = (\mathcal{D}_i)_{i \in \mathbb{Z}}$, such that $\mathfrak{A} \uplus \mathfrak{D}$ is consistent with $\mathcal{T}$, and $\mathcal{T}, \mathfrak{A} \uplus \mathfrak{D}, n \models \phi$. The solution $\mathfrak{D}$ is called:*

– $\preceq_e$-***minimal*** iff *for every solution $\mathfrak{D}'$, if $\mathfrak{D} \models \mathfrak{D}'$ then $\mathfrak{D}' \models \mathfrak{D}$,*
– $\preceq_b$-***minimal*** iff *for every solution $\mathfrak{D}'$, if $\mathcal{T}, \mathfrak{A} \uplus \mathfrak{D} \models \mathfrak{D}'$ then $\mathcal{T}, \mathfrak{A} \uplus \mathfrak{D}' \models \mathfrak{D}$,*
– $\preceq_s$-***minimal*** iff *for every solution $\mathfrak{D}'$, if $\mathfrak{D}' \rightharpoonup_n \mathfrak{D}$ then $\mathfrak{D} = \mathfrak{D}'$.*

In the remainder of this paper, we assume w.l.o.g. that for every TQA problem $(\mathcal{T}, \mathfrak{A}, \phi, n)$, $n = 0$, and write $(\mathcal{T}, \mathfrak{A}, \phi)$ for short. Intuitively, the pair $(\mathcal{T}, \mathfrak{A})$ represents the background knowledge for the abductive inference over the TQL formula ϕ. The finiteness of $\mathfrak{A}$ is one of the necessary conditions to ensure that the space of abductive solutions can be made finite.

As usually in the context of abductive reasoning, we employ several minimality criteria which help to reduce the solution space to a computationally manageable level. The first two are generalizations of criteria known in the classical, atemporal abduction. Intuitively, $\preceq_e$-minimality (for *e*ntailment) places the precedence over solutions which are logically weakest — they assume the least possible data in every given state — irrespectively of the background knowledge. The $\preceq_b$-minimality (for entailment w.r.t. *b*ackground knowledge) takes also into account the assumed TBox and A-sequence. Observe that $\preceq_b$-minimality is strictly stronger than $\preceq_e$-minimality, i.e., whenever a solution $\mathcal{D}$ is $\preceq_b$-minimal it must be $\preceq_e$-minimal, while the converse does not hold in general. Note that

	...	-4	-3	-2	-1	0
$\mathfrak{A}$:			$\mathit{HeavyRainIn}(l)$	$\mathit{tempIn}(t, l)$		
$\mathfrak{D}_1$:				$\mathit{below\_0}(t)$		
$\mathfrak{D}_2$:			$\mathit{RainIn}(l)$	$\mathit{below\_0}(t)$		
$\mathfrak{D}_3$:				$\mathit{HeavyRainIn}(l)$	$\mathit{tempIn}(t, l)$ $\mathit{below\_0}(t)$	
$\mathfrak{D}_4$:		$\mathit{RainIn}(l)$	$\mathit{tempIn}(t, l)$ $\mathit{below\_0}(t)$			

Table 1. The A-sequence $\mathfrak{A}$ and the solutions $\mathfrak{D}_1$-$\mathfrak{D}_4$ to $(\mathcal{T}, \mathfrak{A}, \phi)$, for $\mathcal{T} = \{\mathit{HeavyRainIn} \sqsubseteq \mathit{RainIn}\}$.

whenever a problem has a solution at all, it must have a $\preceq_b$-minimal (and thus an $\preceq_e$-minimal) solution. The $\preceq_s$-minimality criterion (for *s*tructure) is a novel one, tailored specifically for abduction problems whose solutions are sequential structures. It ensures that the solution $\mathfrak{D}$ has no redundant subsequences. To rephrase it, $\mathfrak{D}$ is not minimal in the sense of $\preceq_s$ whenever one can obtain a distinct solution by simply removing some ABoxes from $\mathfrak{D}$ — somewhere to the left or to the right from the fixed point n. In general, the abductive procedures developed in the next section are complete w.r.t. $\preceq_s$- and $\preceq_e$-minimal solutions, while $\preceq_b$-minimality can be optionally used in certain cases to ease the computation.

For a more intuitive illustration of the setup and the employed minimality criteria, we consider a TQA problem $\mathit{\Omega} = (\mathcal{T}, \mathfrak{A}, \phi)$, with $\mathcal{T} = \{\mathit{HeavyRainIn} \sqsubseteq \mathit{RainIn}\}$, $\mathfrak{A}$ as included in Table 1 and ϕ as defined in the beginning of this section. Table 1 presents several solutions to $\mathit{\Omega}$ at time 0. Note, that all empty and hidden cells in the table are empty ABoxes. Solutions $\mathfrak{D}_1$ and $\mathfrak{D}_3$ are both $\preceq_s$- and $\preceq_e$-minimal. $\mathfrak{D}_3$ is not $\preceq_b$-minimal, as replacing $\mathit{HeavyRainIn}(l)$ with its consequence $\mathit{RainIn}(l)$ results in a logically weaker solution w.r.t. $\mathcal{T}$. Further, solution $\mathfrak{D}_2$ is not $\preceq_e$-minimal, as the assertion $\mathit{RainIn}(l)$ is not in fact necessary for the query to be entailed. Finally, solution $\mathfrak{D}_4$ is not $\preceq_s$-minimal. Observe that by dropping ABoxes $\mathcal{D}_{-2}$ and $\mathcal{D}_{-1}$, and "shifting" $\mathcal{D}_{-4}$ and $\mathcal{D}_{-3}$ to the right by two time points, we obtain a non-equivalent $\preceq_s$-minimal solution.

4 Complexity Analysis

In this section, we study the combined complexity of different variants of the temporal query abduction problem. The proofs are included in the appendix. Note that "*recognition*" results, w.r.t., a minimality criterion, signals that the underlying decision procedure is complete but not necessarily sound, i.e. we ensure that all minimal solutions are found by the procedure, but it might be still necessary to filter out some non-minimal ones which are also included in the outcome. A "*computation*" result implies soundness as well [7].

We start by addressing ABox abduction, i.e., the problem of abducing a minimal ABox ensuring entailment and non-entailment of selected CQs at a single time point.

Definition 5 (ABox abduction). *An ABox abduction problem is a tuple $\Omega = (\mathcal{T}, \mathcal{A}, P, N)$, where $\mathcal{T}$ is a TBox, $\mathcal{A}$ an ABox, and $P, N \subseteq \mathcal{Q}_\Sigma$ are sets of boolean CQs. An ABox $\mathcal{D}$ is called a solution to problem Ω* iff *$\mathcal{A} \cup \mathcal{D}$ is consistent with $\mathcal{T}$ and:*

1. *$\mathcal{T}, \mathcal{A} \cup \mathcal{D} \models [q]$, for every $q \in P$,*
2. *$\mathcal{T}, \mathcal{A} \cup \mathcal{D} \models \neg[q]$, for every $q \in N$.*

Note, that $\preceq_e$- and $\preceq_b$-minimality criteria transfer immediately from Definition 4, on considering a single ABox as an A-sequence with exactly one element. The $\preceq_s$-minimality does not apply in the context of ABox abduction. The results obtained here rest on and extend some of those presented in [7].

Lemma 1 (Solving ABox abduction problems). *Let Ω be an ABox abduction problem and $\mathcal{D}$ an $\preceq_e$-minimal solution to Ω. Then:*

1. *computing $\mathcal{D}$ for $\Omega = (\mathcal{T}, \emptyset, P, \emptyset)$ is in* PTime*, if $\mathcal{T} = \emptyset$ or $\mathcal{D}$ is $\preceq_b$-minimal,*
2. *recognizing $\mathcal{D}$ for $\Omega = (\mathcal{T}, \mathcal{A}, P, \emptyset)$ is* NP*-complete, if $\mathcal{T} \neq \emptyset$ or $\mathcal{A} \neq \emptyset$,*
3. *computing $\mathcal{D}$ for $\Omega = (\mathcal{T}, \mathcal{A}, P, N)$ is* DP*-complete, if $P \neq \emptyset$ and $N \neq \emptyset$, even when $\mathcal{A} = \emptyset$ and irrespective of deciding $\preceq_b$-minimality,*

where $\mathcal{D}$ is fixed up to renaming individuals in the included ABox assertions.

The PTime result in the first case follows by observing that the addressed ABox abduction problems can be solved immediately by grounding the conjuncts of the CQs. Solving the second type of problems might involve NP-complete CQ entailment checks and/or a nondeterministic choice from an exponential number of queries in the FO rewriting of a CQ. For the last case, recall that DP denotes the intersection of the classes of NP and coNP problems. The result is due to the simultaneous presence of positive and negative CQs, which requires entailment and non-entailment checks, with the latter in coNP.

Next, we focus on proper TQA problems in TQL. The central challenge to be addressed is that solutions to such problems are in principle of infinite length, which makes their computation generally impossible in finite time. However, we are able to identify certain finite structures which can be unambiguously unfolded into the corresponding A-sequences. Thus, rather than searching for A-sequences directly, we focus on finding their finite representations, called *A-structures*.

Definition 6 (A-structures). *An A-structure is a tuple $\mathfrak{S} = (S, \mathcal{S}_0, \rightarrow)$, where $S = S^+ \cup S^-$ is a finite set of ABoxes, $\mathcal{S}_0 \in S^+ \cap S^-$ is the initial ABox, and $\rightarrow$ is a successor function, such that $\rightarrow: S^+ \mapsto S^+$ and $\rightarrow: S^- \mapsto S^-$. The* unfolding *of $\mathfrak{S}$ is an A-sequence $\ldots, \mathcal{S}_{j-1}, \mathcal{S}_j, \ldots, \mathcal{S}_0, \ldots, \mathcal{S}_i, \mathcal{S}_{i+1}, \ldots$, where for every $i \geq 0$, $\mathcal{S}_i \rightarrow \mathcal{S}_{i+1}$, for some $\mathcal{S}_{i+1} \in S^+$, and for every $j \leq 0$, $\mathcal{S}_j \rightarrow \mathcal{S}_{j-1}$, for some $\mathcal{S}_{j-1} \in S^-$.*

The key to the abductive algorithms we develop here is ensuring existence of an upper bound on the size of the A-structures that are to be found. Technically, the proofs rest on the construction of so-called *quasimodels*, which link A-structures with the input abductive problems. Intuitively, a quasimodel $\boldsymbol{s} = (s_i)_{i \in \mathbb{Z}}$ is an abstraction of an infinite model satisfying the query. Each s_i-th element $(t_i, \mathcal{B}(t_i))$ in that sequence consists of the set t_i of subformulas of ϕ that must be satisfied in i and the minimal ABox $\mathcal{B}(t_i)$ satisfying all positive and negative occurrences of CQs at i, i.e., $[q], \neg[q] \in t_i$. Particularly instrumental are special quasimodels called ultimately periodic, comprised by two infinite subsequences (one future, one past), where each subsequence consists of a finite initial sequence called the head, followed by an infinite repetition of some terminal subsequence of the head, called the period. We show that every $\preceq_e$- and $\preceq_s$-minimal solution to a TQA problem corresponds to an ultimately periodic quasimodel, which can be further associated with an A-structure of a particular size, linear in the length of the heads of the two sequences comprising the quasimodel.

For TQA over TQL formulas the relevant A-structures consist of at most exponentially many states in the size of the given abduction problem. This resonates closely with the "small model" property of LTL, which rests on similarly defined bounds [23]. Recall that by $\ell(\dagger)$ we denote the total size of the input $\dagger$.

Lemma 2 (A-sequence vs. A-structure). *Let $\mathfrak{D}$ be an $\preceq_e$- and $\preceq_s$-minimal solution to a TQA problem $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$, where $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ and ϕ is a* TQL *formula. Then there exists an A-structure $\mathfrak{S}$ whose unfolding is $\mathfrak{D}$, such that $|S| = f(\ell(\Omega))$, for some function $f(x) \in O(2^x)$.*

The basic algorithm which recognizes $\preceq_e$- and $\preceq_s$-minimal solutions to TQA problems is an adaptation of Sistla and Clarke's decision procedure for LTL [23]. In principle, the underlying computation model has to be changed from finite-state automata to finite-state transducers, i.e., Turing machines using additional write-only output tapes, as a recognized solution needs to be effectively presented. This revision, however, does not affect the complexity of the algorithm, which remains PSpace-complete, irrespectively of the possibly exponential size of solutions.

Theorem 1 (Recognizing TQA solutions). *Recognizing an $\preceq_e$- and $\preceq_s$-minimal solution to a TQA problem $(\mathcal{T}, \mathfrak{A}, \phi)$, where ϕ is a* TQL *formula, is* PSpace*-complete.*

As the hardness result transfers from the satisfiability problem in the underlying LTL, it is easy to see that the result holds even for pure-future (only U operator) or pure-past TQL formulas. In case of $\text{TQL}^{\mathsf{F},\mathsf{P}}$ and TQL^{+} we are able to show that the upper bound on the size of the relevant A-structures is smaller — in fact, linear in the size of the input.

Lemma 3 (A-sequence vs. A-structure for $\text{TQL}^{\mathsf{F},\mathsf{P}}, \text{TQL}^{+}$). *Let $\mathfrak{D}$ be an $\preceq_e$- and $\preceq_s$-minimal solution to a TQA problem $(\mathcal{T}, \mathfrak{A}, \phi)$, where $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ and ϕ is a $\text{TQL}^{\mathsf{F},\mathsf{P}}$ or TQL^{+} formula. Then there exists an A-structure $\mathfrak{S}$ whose unfolding is $\mathfrak{D}$, such that $|S| \leq f(\ell(\phi))$, for some $f(x) \in O(x)$.*

Given the linear size of the solutions, the worst case complexity of recognizing TQA solutions for $\text{TQL}^{\mathsf{F},\mathsf{P}}$ drops to DP. In this case, it is sufficient to guess a linearly long head of a candidate quasimodel and verify it satisfies all the necessary structural conditions. As states in the quasimodel can contain positive and negative occurrences of CQs, the abduction of the respective minimal ABoxes is DP-complete.

Theorem 2 (Recognizing TQA solutions for $\text{TQL}^{\mathsf{F},\mathsf{P}}$**).** *Recognizing an* $\preceq_e$*- and* $\preceq_s$*-minimal solution to a TQA problem* $(\mathcal{T}, \mathfrak{A}, \phi)$*, where* ϕ *is a* $\text{TQL}^{\mathsf{F},\mathsf{P}}$ *formula, is* DP*-complete.*

In case of TQL^{+}, the complexity of abductive reasoning is even smaller, in fact NP-complete, as no negative CQs have to be considered. Reducing the TQL language further down to $\text{TQL}^{\mathsf{F},\mathsf{P},+}$ does not yield any additional gain, even when $\preceq_b$-minimality is considered and the A-sequence $\mathfrak{A}$ is empty. This is a consequence of the non-determinism involved in choosing the order in which U-/S-formulas are fulfilled in the consecutive states. In the worst case, all permutations must be considered, which enables reduction from the NP-hard Hamiltonian path problem.

Theorem 3 (Recognizing TQA solutions for $\text{TQL}^{+}, \text{TQL}^{\mathsf{F},\mathsf{P},+}$**).** *Recognizing a* $\preceq_e$*- and* $\preceq_s$*-minimal solution to a TQA problem* $(\mathcal{T}, \mathfrak{A}, \phi)$*, where* ϕ *is a* TQL^{+} *or* $\text{TQL}^{\mathsf{F},\mathsf{P},+}$ *formula, is* NP*-complete. The result holds even for* $\preceq_b$*-minimal solutions and when* $\mathfrak{A} = \emptyset$*.*

Note that in most cases computing TQA solutions, as opposed to recognizing them, is bound to be of a higher complexity due to the necessity of conducting pairwise comparisons between exponentially many alternatives.

The findings reported above, neatly reflect the modular character of TQL, which allows for handling the embedded CQs largely independently from reasoning about the temporal dimension of the queries. In case of TQL entailment, this feature suggests a universal way of defining the upper complexity bound in different fragments of the language [13]. The bound is implied by the generic algorithm consisting of any (standard) decision procedure for the temporal language, augmented with an oracle deciding entailment/non-entailment of the CQs. Whenever the underlying LTL is in PSpace, then TQL entailment must be in $\text{PSpace}^{\text{DP}} = \text{PSpace}$. In $\text{TQL}^{\mathsf{F},\mathsf{P}}$ the same argument leads to a procedure in NP^{DP}=DP. Finally in $\text{TQL}^{\mathsf{F},\mathsf{P},+}$, where CQs occur only in positive form, we obtain NP^{NP}=NP bound. These results clearly mirror the identified bounds for TQA problems in the corresponding fragments.

The analysis conducted in this section shows that TQA problems are computationally hard in general, but can be made easier by progressively simplifying the assumed setting. Notably, by restricting the expressiveness of temporal operators and eliminating negation from the underlying TQL, the complexity of reasoning can be reduced from PSpace- to NP-complete. The remaining non-determinism, warranting NP-hardness, can be mostly attributed to the size of FO rewritings of CQs and the number of alternative orders in which U/S-subformulas are to be fulfilled over time. Can these too be tamed granting an even lower complexity? Most likely, yes. We suspect that by considering $\preceq_b$-minimal solutions and allowing only formulas whose structure unambiguously determines the order of fulfilment of U/S-subformulas, the combined complexity of prediction and explanation should drop further to PTime.

5 Conclusions

In this paper, we have defined a novel problem of *temporal query abduction* in the context of data represented in DL-*Lite*. A number of complexity results, which we have delivered for different restricted fragments of the studied setting, imply concrete ways of constraining the problem in order to render abductive reasoning more feasible in practice. As processing temporal semantic data becomes an increasingly important task in many applications, understanding and being able to operationalize such constraints is crucial for efficient implementations.

More generally, we believe that the use of TQL-like queries for representing and reasoning about temporal data patterns and their correlations defines a highly promising approach to bridging the gap between the semantic perspective on temporal data and the statistical view, endorsed in the data mining field. As initiated in [16], we intend to further the study of forms of reasoning which could naturally benefit from existence of such a link, such as prediction and explanation over time series and streaming semantic data.

References

1. Baader, F., Calvanese, D., Mcguinness, D.L., Nardi, D., Patel-Schneider, P.F.: The description logic handbook: theory, implementation, and applications. Cambridge University Press (2003)
2. Batsakis, S., Stravoskoufos, K., Petrakis, E.G.M.: Temporal reasoning for supporting temporal queries in OWL 2.0. In: Proc. of the Int. Conf. on Knowledge-based and Intelligent Information and Engineering Systems (KES-11) (2011)
3. Bienvenu, M.: Complexity of abduction in the EL family of lightweight description logics. In: Proc. of the International Conference on Principles of Knowledge Representation and Reasoning (KR-08) (2008)
4. Borgwardt, S., Lippmann, M., Thost, V.: Temporal query answering in the description logic DL-Lite. In: Proc. of the International Symposium on Frontiers of Combining Systems (FroCoS-13) (2013)
5. Brusoni, V., Console, L., Terenziani, P., Dupr, D.: An efficient algorithm for temporal abduction. In: Proc. of Advances in Artificial Intelligence (AI*IA-97) (1997)
6. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tractable reasoning and efficient query answering in description logics: The DL-Lite family. J. of Automated Reasoning 39(3), 385–429 (2007)
7. Calvanese, D., Ortiz, M., Simkus, M., Stefanoni, G.: The complexity of explaining negative query answers in DL-Lite. In: Proc. of the International Conference on the Principles of Knowledge Representation and Reasoning (KR-12) (2012)

8. Colucci, S., Noia, T.D., Sciascio, E.D., Donini, F.M., Mongiello, M.: A uniform tableaux-based approach to concept abduction and contraction in ALN. In: Proc. of the International Workshop on Description Logics (DL-04) (2004)
9. Elsenbroich, C., Kutz, O., Sattler, U.: A case for abductive reasoning over ontologies. In: Proc. of the International Workshop on OWL: Experiences and Directions (OWLED-06) (2006)
10. Espinosa Peraldi, S., Kaya, A., Melzer, S., Möller, R., Wessel, M.: Multimedia interpretation as abduction. In: Proc. of the International Workshop on Description Logics (DL-07) (2007)
11. Gabbay, D.: The declarative past and imperative future: Executable temporal logic for interactive systems. In: Proc. of the Conference on Temporal Logic in Specification (TLS-87) (1987)
12. Grandi, F.: T-SPARQL: a TSQL2-like temporal query language for RDF. In: Proc. of the International Workshop on Querying Graph Structured Data (2010)
13. Gutiérrez-Basulto, V., Klarman, S.: Towards a unifying approach to representing and querying temporal data in description logics. In: Proc. of the Int. Conf. on Web Reasoning and Rule Systems (RR-12) (2012)
14. Halland, K., Britz, K.: Naive ABox abduction in ALC using a DL tableau. In: Proc. of the International Workshop on Description Logics (DL-12) (2012)
15. Klarman, S., Endriss, U., Schlobach, S.: ABox abduction in the description logic $\mathcal{ALC}$. Journal of Automated Reasoning 46, 43–80 (2011)
16. Klarman, S., Meyer, T.: Prediction and explanation over DL-*Lite* data streams. In: Proc. of the International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR-19) (2013)
17. Klarman, S., Meyer, T.: Complexity of temporal query abduction in DL-*Lite*. Tech. rep., CAIR, UKZN/CSIR Meraka. `http://klarman.synthasite.com/resources/KlaMeyDL14.pdf` (2014)
18. Lambrix, P., Wei-Kleiner, F., Dragisic, Z., Ivanova, V.: Repairing missing is-a structure in ontologies is an abductive reasoning problem. In: Proc. of the International Workshop on Debugging Ontologies and Ontology Mappings (WoDOOM-13) (2013)
19. Lutz, C., Toman, D., Wolter, F.: Conjunctive query answering in the description logic EL using a relational database system. In: Proc. of the International Joint Conference on Artifical Intelligence (IJCAI-09) (2009)
20. Motik, B.: Representing and querying validity time in RDF and OWL: A logic-based approach. Web Semantics: Science, Services and Agents on the World Wide Web 12-13, 3–21 (2012)
21. Pan, J.Z., Du, J., Qi, G., Shen, Y.D.: Towards practical ABox abduction in large description logic ontologies. International Journal on Semantic Web and Information Systems 8(2), 1–33 (2012)
22. Ribeiro, C., Porto, A.: Abduction in temporal reasoning. In: Proc. of the International Conference on Temporal Logic (ICTL-94) (1994)
23. Sistla, A.P., Clarke, E.M.: The complexity of propositional linear temporal logics. Journal of ACM 32(3), 733–749 (1985)