Prediction and Explanation over DL-*Lite* Data Streams

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. Stream reasoning is an emerging research area focusing on the development of reasoning techniques applicable to streams of rapidly changing, semantically enhanced data. In this paper, we consider data represented in Description Logics from the popular DL-*Lite* family, and study the logic foundations of prediction and explanation over DL-*Lite* data streams, i.e., reasoning from finite segments of streaming data to conjectures about the content of the streams in the future or in the past. We propose a novel formalization of the problem based on temporal "past-future" rules, grounded in Temporal Query Language. Such rules can naturally accommodate complex data association patterns, which are typically discovered through data mining processes, with logical and temporal constraints of varying expressiveness. Further, we analyse the computational complexity of reasoning with rules expressed in different fragments of the temporal language. As a result, we draw precise demarcation lines between NP-, DP- and PSpace-complete variants of our setting and, consequently, suggest relevant restrictions rendering prediction and explanation more feasible in practice.

1 Introduction

A data stream is a temporally ordered collection of data, representing the flow of information through a certain channel over time [1]. Semantic applications generating and consuming such streams of rapidly changing data are becoming increasingly common, with domains ranging through scientific, medical, financial, urban, and many others. As has been argued by many authors, the shift of the paradigm from traditional, static data to streaming information requires deep revisions and advancements in the area of automated reasoning. On the one hand, the capacity and velocity of data streams present a serious technological challenge for the existing reasoning systems, tailored towards static data models and softer latency requirements. On the other one, the real-time and real-world nature of streaming information encourages investigations into novel forms of reasoning, going beyond the basic, deductive query answering — forms, which could support the construction of versatile analytical tools for enhancing the understanding and utilization of knowledge conveyed in data streams [2,3]. This latter research agenda motivates directly our presented work.

In this paper, we study the logic foundations of two non-deductive types of inference over data streams: *prediction* and *explanation*, i.e., reasoning from finite segments of streaming data to conjectures about the content of the streams in the future and in the past. Thus defined notions of prediction and explanation are variations of their well-established analogs in philosophy of science, where they are often related to the classical problem of causality.[1] There, to predict is to identify the expected effects of existing causes, while to explain — inversely — to find possible causes of the observed effects [4]. In systems managing real-time information, prediction is of major importance as an inference guiding decision making processes based on the currently available data. Meanwhile, explanation is pivotal to comprehending the situation which underlies and justifies the observed data, which often requires procuring the relevant chain of circumstances leading to it or abstracting the data into higher-level knowledge. Both modes of inference are essential for achieving situation awareness in a real-time information system [5]. Although prediction (and to a lesser degree explanation) has been addressed in the context of streaming data, the focus of the relevant work lies predominantly on the data mining level, i.e., on the methodology of learning the association patterns occurring in the data and extrapolating them via statistical techniques to yet unobserved data [3,6]. On the contrary, virtually no attention has been given to predictive and explanatory reasoning in its strictly logical sense, as a symbolic inference, on the knowledge representation level. This is a critical gap whenever streams of semantically rich data are considered, as in such scenarios bridging the statistical and semantic view on the data is instrumental to designing robust reasoning techniques. To the best of our knowledge, in this work we present the first insights and results on logical and computational aspects of prediction and explanation over semantic data streams.

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 DLs from the popular DL-*Lite* family [7]. Further, we define a special type of temporal "past-future" rules, grounded in Temporal Query Language [8]. Such rules can naturally accommodate complex data association patterns, identified in the data mining phase, with logical and temporal constraints of varying expressiveness. Based on this foundation, we propose a novel formalization of the two studied types of inference, as abduction of a data sequence satisfying the consequent or, respectively, the antecedent of a temporal rule. We analyse the computational complexity of such tasks over rules expressed in different fragments of the temporal language, and as a result, we draw precise demarcation lines between NP-, DP- and PSPACE-complete variants of the problem. Building on these findings, we discuss relevant restrictions to the prediction and explanation tasks which can render the reasoning feasible in practice.

The paper is organized as follows. In the next section, we recap preliminaries of DLs and conjunctive query answering. In Section 3, we systematically introduce all temporal components of the framework, including data streams, Temporal Query Language and temporal rules. Then, in Section 4, we define prediction

[1] See `http://plato.stanford.edu/entries/scientific-explanation/`.

and explanation and motivate our proposal. In Section 5, we present the complexity results and, further, discuss their consequences on the main problem. The proofs of the results are included in the appendix. An overview of related work and concluding remarks are presented in the last two sections.

2 Preliminaries

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 [9]. 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 [9]. 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 grounded whenever $\mathsf{avar}(q) = \emptyset$. A grounded 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 $\mathsf{cert}(\mathsf{q}, \mathcal{T}, \mathcal{A})$. 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 [7], such as DL-$Lite_{\mathcal{R}}$, DL-$Lite_{\mathcal{F}}$ or DL-$Lite_{\mathcal{A}}$, underlying the OWL 2 QL ontology language profile [2], for which CQs enjoy the so-called first-order rewritability property, defined as follows.

Definition 1 (FO rewritability [7]). *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 $\sigma \in \mathsf{cert}(\mathsf{q}, \mathcal{T}, \mathcal{A})$* 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.*

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

Recall, that given $\mathcal{T}$ in any of such DLs and a grounded 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 of the size linear in $\ell(q)$ and is such that $\mathcal{T} \cup \{q'\} \models q$. 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 [7].

Regardless of this default focus, many of the results presented here can be naturally extended to other DLs exhibiting similar characteristics, such as other members of the DL-*Lite* family or logics in the $\mathcal{EL}$ family [10].

3 Temporal data and queries

We consider a discrete, linear *flow of time* $(\mathbb{Z}, <)$, with integers representing *time instants* 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 $I^- \leq I^+ \in \mathbb{Z} \cup \{-\infty, +\infty\}$ denote the beginning and the end of I, respectively. We assume that $\mathbb{N} = [0, +\infty]$.

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 temporally w.r.t. the underlying time flow. 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:

- $\mathfrak{A} \subseteq \mathfrak{B}$ ($\mathfrak{A} = \mathfrak{B}$) *iff* $I \subseteq J$ ($I = J$) and $\mathcal{A}_i = \mathcal{B}_i$ for every $i \in I$,
- $\mathcal{T}, \mathfrak{A} \models \mathfrak{B}$ ($\mathfrak{A} \models \mathfrak{B}$) *iff* $J \subseteq I$ and $\mathcal{T}, \mathcal{A}_i \models \mathcal{B}_i$ ($\mathcal{A}_i \models \mathcal{B}_i$) for every $i \in J$,
- $\mathfrak{A} \rightharpoonup \mathfrak{B}$ *iff* there exists a mapping $f : I \mapsto J$, such that:
  - $i < j$ *iff* $f(i) < f(j)$, for every $i, j \in I$,
  - $\mathcal{A}_i = \mathcal{B}_{f(i)}$, for every $i \in I$,
- $\mathfrak{A} \uplus \mathfrak{B}$, 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 $i \in I \cap J$,
- $\mathfrak{A}_{\leq n}$ ($\mathfrak{A}_{\geq n}$), for $n \in I$, to denote the A-sequence $(\mathcal{A}_i)_{i \in I'} \subseteq \mathfrak{A}$, such that $I' = [I^-, n]$ ($I' = [n, I^+]$).

The notion of data stream adopted here specializes that of ontology stream, as introduced in [11], by considering temporal variability only on the data (ABox) level, while prohibiting changes on the ontology (TBox) level.

Definition 3 (Data stream). *A* data stream *under a TBox $\mathcal{T}$ is an A-sequence $\mathfrak{A} = (\mathcal{A}_i)_{i \in \mathbb{Z}}$ consistent with $\mathcal{T}$, with a designated subsequence $\mathfrak{A}_\omega \subseteq \mathfrak{A}$, called the* recorded segment *of $\mathfrak{A}$, where ω is a finite interval over $\mathbb{Z}$. For the* current time *$n \in \mathbb{Z}$, we call $\mathfrak{A}_{\leq n}$ the* past*, and $\mathfrak{A}_{\geq n}$ the* future *of $\mathfrak{A}$.*

In full generality, a data stream is then an infinite sequence of datasets consistent with a fixed TBox. Obviously, in practical scenarios, one can effectively know and manage only a finite fragment of the past of a given stream, while remaining agnostic about its future. What we call above the recorded segment of $\mathfrak{A}$ is precisely this finite, accessible portion of the stream.

Next, we recall a variant of Temporal Query Language, proposed in [8], to be used for accessing data streams. It is a lightweight combination of Linear Temporal Logic (LTL) [12] with CQs, where CQs are embedded in the temporal language using the epistemic semantics.

Definition 4 (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:*

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

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 grounded *whenever $\mathsf{avar}(\phi) = \emptyset$. The entailment relation for grounded 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:*

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

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 $\mathsf{cert}_i(\phi, \mathcal{T}, \mathfrak{A})$.*

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

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

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:

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

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. Importantly, such lightweight combination of languages allows also for a modular reuse of existing temporal reasoners and highly optimized, efficient query answering engines [8].

LTL with operators U and S, standing for (strict) *until* and *since*, which captures precisely the temporal component of TQL, is known to be expressively complete over $(\mathbb{Z}, <)$ [13]. Apart from the full TQL, in what follows we consider also some of its strict subsets. By $\text{TQL}^{\exists}$ we denote the fragment in which the syntax of U- and S-formulas is restricted to the form $\top\mathsf{U}\phi$ and $\top\mathsf{S}\phi$, where $\top$ is a constant symbol denoting the logical truth. This restriction corresponds to LTL with operators *sometime in the future* and *sometime in the past*, in place of U and S. Further, with TQL^{+} we refer to the positive fragment of TQL, i.e., TQL without the negation operator. Finally, by $\text{TQL}^{\exists,+}$, we denote the intersection of $\text{TQL}^{\exists}$ and TQL^{+}.

Following the temporal separation approach of Gabbay [13], we consider TQL formulas belonging to two disjoint categories:

- ***past-present***: formulas without the operators of type U,
- ***future-present***: formulas without the operators of type S.

By the semantics of TQL, it follows that for any TQL formula ϕ, TBox $\mathcal{T}$, A-sequence $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$, and time point $n \in I$, the equivalences below hold:

- $\mathsf{cert}_n(\phi, \mathcal{T}, \mathfrak{A}) = \mathsf{cert}_n(\phi, \mathcal{T}, \mathfrak{A}_{\leq n})$, whenever ϕ is past-present,
- $\mathsf{cert}_n(\phi, \mathcal{T}, \mathfrak{A}) = \mathsf{cert}_n(\phi, \mathcal{T}, \mathfrak{A}_{\geq n})$, whenever ϕ is future-present.

Given the distinction above, we define the notion of temporal rules, which is closely related to Gabbay's concept of executable temporal logic [13]. Temporal rules straightforwardly embody the "*declarative past–imperative future*" pattern over TQL.

Definition 5 (Temporal rules). *A temporal rule in TQL is an expression of the form:*

$$\psi \Rightarrow \phi$$

where ψ, ϕ are TQL formulas such that ψ is past-present and ϕ is future-present. The rule $\psi \Rightarrow \phi$ is satisfied for a substitution $\varrho = \sigma \cup \sigma'$, for some $\sigma : \mathsf{avar}(\psi) \mapsto \mathsf{N_I}$ and $\sigma' : \mathsf{avar}(\phi) \mapsto \mathsf{N_I}$ agreeing on $\mathsf{avar}(\psi) \cap \mathsf{avar}(\phi)$, over a TBox $\mathcal{T}$ and an A-sequence $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$, at time $n \in I$ iff *$\sigma \in \mathsf{cert}_n(\psi, \mathcal{T}, \mathfrak{A})$ implies $\sigma' \in \mathsf{cert}_n(\phi, \mathcal{T}, \mathfrak{A})$.*

Temporal rules are equipped with well-defined semantics and allow for relatively easy control of the expressiveness-complexity trade-off, due to their close relationship with LTL. They are also a natural formalism for expressing association rules discoverable in time series data by means of various data mining techniques. This sort of association rules typically combine diverse data patterns with logical and temporal constraints [14]. Although lacking some essential probabilistic and real-time features, not present in the basic variants of LTL, temporal rules can arguably provide a robust logic foundation for target learning languages over streaming DL-*Lite* data. As an example, we present a prototypical temporal association rule used in a climate application predicting droughts in certain regions of India [15]

Climate application example: Consider a temporal rule $\psi \Rightarrow \phi$ encoding a correlation between several measurements and weather phenomena occurring in specific geographic locations, in a specific order, known to be a good predictor of drought. The rule is defined by the TQL formulas:

$$\psi = (\neg[\exists y.(\mathit{HeavyRainIn}(y) \land \mathit{locIn}(y, \mathit{north}))] \; \mathsf{S} \; [\exists y, z.(\mathit{SST}(y, \mathit{low}) \land \mathit{NAO}(z, \mathit{high})]) \land [\mathit{locIn}(x, \mathit{northeast})]$$

$$\phi = \top \; \mathsf{U} \; ([\mathit{DroughtIn}(x)] \land ([\mathit{DroughtIn}(x)] \; \mathsf{U} \; [\mathit{SevereDroughtIn}(x)]))$$

It states that if at some point in the past the SST (sea surface temperature) was found out to be low, the NAO (North Atlantic Oscillation) was high, and since then there has been no heavy rain recorded in North province, then at some point in the future there will be drought in x, whenever x is located in Northeast, which will persist until severe drought occurs in x. □

4 Prediction and explanation

By adopting temporal rules as the language of association patterns in streaming data, we are able to formulate very intuitive and clear-cut definitions of prediction and explanation over data streams: *a prediction (explanation) is a possible future (past) of the data stream, which entails the consequent (antecedent) of a temporal rule, given its antecedent (consequent) is entailed by the recorded segment.* This meaning of the two types of inference is schematically depicted in Figure 1 and further made precise in the following two definitions. We consider a data stream $\mathfrak{A}$ under a TBox $\mathcal{T}$, with the recorded segment $\mathfrak{A}_\omega \subseteq \mathfrak{A}$, where ω is a finite interval over $\mathbb{Z}$.

Definition 6 (Prediction). *Let $\psi \Rightarrow \phi$ be a temporal rule and $\sigma \in \mathsf{cert}_n(\psi, \mathcal{T}, \mathfrak{A}_\omega)$, for a time $n \in \omega$. A* prediction *at n from $\psi \Rightarrow \phi$ and σ over $\mathcal{T}, \mathfrak{A}_\omega$ is an A-sequence $\mathfrak{D} = (\mathcal{D}_i)_{i \in [n, +\infty]}$ such that $\sigma' \in \mathsf{cert}_n(\phi, \mathcal{T}, \mathfrak{A}_\omega \uplus \mathfrak{D})$, for some σ' agreeing with σ on $\mathsf{avar}(\psi) \cap \mathsf{avar}(\phi)$.*

Definition 7 (Explanation). *Let $\psi \Rightarrow \phi$ be a temporal rule and $\sigma \in \mathsf{cert}_n(\phi, \mathcal{T}, \mathfrak{A}_\omega)$, for a time $n \in \omega$. An* explanation *of σ at n based on $\psi \Rightarrow \phi$ is an A-sequence $\mathfrak{D} = (\mathcal{D}_i)_{i \in [-\infty, n]}$ such that $\sigma' \in \mathsf{cert}_n(\psi, \mathcal{T}, \mathfrak{A}_\omega \uplus \mathfrak{D})$ for some σ' agreeing with σ on $\mathsf{avar}(\psi) \cap \mathsf{avar}(\phi)$.*

From a high-level perspective, prediction and explanation are classifiable as strictly different types of inference in that the former is deductive (following from

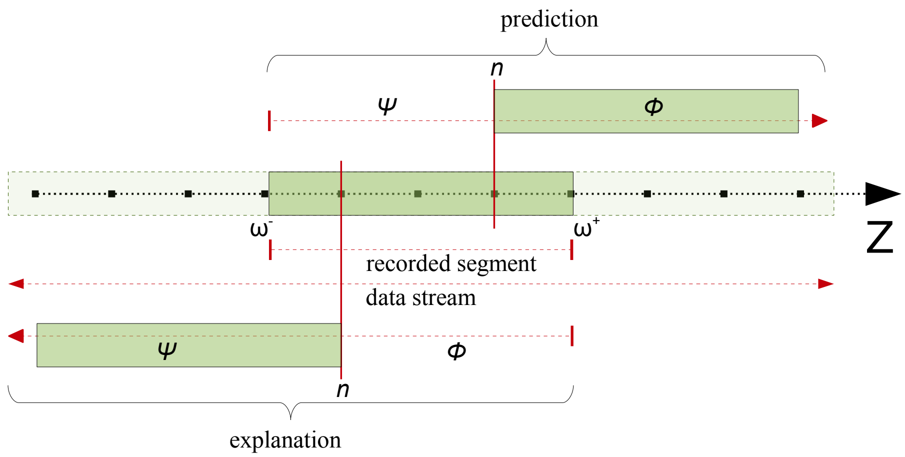

Fig. 1. Prediction and explanation over data streams.

the antecedent to the consequent), while the latter abductive (from the consequent to the antecedent) [4]. Technically, however, it is strikingly apparent that the gist of both inferences is essentially the same and comes down to solving two identical subproblems: 1) verifying that a certain TQL formula (ψ in prediction and ϕ in explantation) is entailed by the recorded segment, thus triggering the particular inference, and 2) finding an A-sequence which entails the second TQL formula in the temporal rule (ϕ in prediction and ψ in explantation). As far as the former task, reducible to deductive entailment, is relatively well-understood, and hence is only shortly addressed in the next section, the latter has not yet been formulated in the literature, and is the central problem studied in the remainder of this paper. The problem has a strongly abductive flavour and is conceptualized here based on the nomenclature coined in [16,17,18].

Definition 8 (A-sequence abduction). *An* A-sequence abduction problem *is a tuple* $(\mathcal{T}, \mathfrak{A}, \phi)$*, where* $\mathcal{T}$ *is a TBox,* $\mathfrak{A} = (\mathcal{A}_i)_{i\in I}$ *is an A-sequence, for some* $I^-, I^+ \in \mathbb{Z}$*, and* ϕ *is a grounded future-present (resp. past-present) TQL formula. A* solution *to* $(\mathcal{T}, \mathfrak{A}, \phi)$ *is an A-sequence* $\mathfrak{D} = (\mathcal{D}_i)_{i\in J}$ *with* $J = [I^-, +\infty]$ *(resp.* $J = [-\infty, I^+]$*), such that* $\mathfrak{A} \uplus \mathfrak{D}$ *is consistent with* $\mathcal{T}$*, and* $\mathcal{T}, \mathfrak{A} \uplus \mathfrak{D}, 0 \models \phi$*. 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* $\mathfrak{D}' \rightharpoonup \mathfrak{D}$ *then* $\mathfrak{D} = \mathfrak{D}'$*.*

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

...	−2	−1	0	1	...
	$locIn(l_1, north)$ $locIn(l_2, northeast)$	$locIn(l_1, north)$ $locIn(l_2, northeast)$ $DroughtIn(l_2)$ $SST(m_1, low)$ $NAO(m_2, high)$	$locIn(l_1, north)$ $locIn(l_2, northeast)$ $SevereDroughtIn(l_2)$ $RainIn(l_1)$	$locIn(l_1, north)$ $locIn(l_2, northeast)$	

Table 1. Data stream in the climate application example.

	1	2	3	4	...
$\mathfrak{D}_1$:		$DroughtIn(l_2)$	$SevereDroughtIn(l_2)$		
$\mathfrak{D}_2$:	$RainIn(l_1)$	$DroughtIn(l_2)$	$SevereDroughtIn(l_2)$		
$\mathfrak{D}_3$:		$SevereDroughtIn(l_2)$	$SevereDroughtIn(l_2)$		
$\mathfrak{D}_4$:		$DroughtIn(l_2)$	$DroughtIn(l_2)$	$SevereDroughtIn(l_2)$	

	...	−5	−4	−3	−2
$\mathfrak{D}_5$:				$SST(m_1, low)$ $NAO(m_2, high)$	
$\mathfrak{D}_6$:				$SST(m_1, low)$ $NAO(m_2, high)$	$locIn(l_2, northeast)$
$\mathfrak{D}_7$:			$SST(m_1, low)$ $NAO(m_2, high)$		

Table 2. Predictions (up) and explanations (down) in the climate application example.

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 ABox. 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 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 the identified sequence $\mathfrak{D}$ has no redundant subsequences. To rephrase it, $\mathfrak{D}$ is not minimal in the sense of $\preceq_s$ whenever one can obtain a solution distinct from $\mathfrak{D}$ simply by removing some ABoxes from $\mathfrak{D}$.

The minimality criteria, discussed above, are consequently applied to predictions and explanations. In fact, the abductive procedures developed in the next section are complete for $\preceq_s$- and $\preceq_e$-minimal solutions, and in practice, we also tend to favor $\preceq_b$-minimal solutions, as more basic. For a more intuitive illustration of the two tasks and the minimality criteria we elaborate further on the climate application scenario, introduced in the previous section.

Climate application example cntd.: Let $\psi \Rightarrow \phi$ be the temporal rule as before, grounded with the substitution $\sigma = \{x \mapsto l_2\}$. Consider TBox $\mathcal{T} = \{\mathit{SevereDroughtIn} \sqsubseteq \mathit{DraughtIn},\ \mathit{HeavyRainIn} \sqsubseteq \mathit{RainIn}\}$ and data stream $\mathfrak{A}$ with the recorded segment $\mathfrak{A}_\omega \subseteq \mathfrak{A}$, where $\omega = [-2, 1]$, defined as in Table 1. Table 2 presents several predictions from $\psi \Rightarrow \phi$ and σ, at time 1 ($\mathfrak{D}_1$-$\mathfrak{D}_4$) and explanations of σ based on $\psi \Rightarrow \phi$, at time -2 ($\mathfrak{D}_5$-$\mathfrak{D}_7$). To put equivalently, these are possible solutions to the A-sequence abduction problems $(\mathcal{T}, \mathcal{A}_1, \phi)$ and $(\mathcal{T}, \mathcal{A}_{-2}, \psi)$, respectively. Note, that all empty and hidden cells in the table are empty ABoxes. Observe that $\mathfrak{D}_1$ and $\mathfrak{D}_5$ are both $\preceq_s$- and $\preceq_b$-minimal. Equivalently, Solutions $\mathfrak{D}_2$ and $\mathfrak{D}_6$ are still $\preceq_s$-minimal but not $\preceq_e$-minimal, and hence not $\preceq_b$-minimal either. In the case of $\mathfrak{D}_2$ axiom $\mathit{RainIn}(l_1) \in \mathcal{D}_1$, although not undermining the prediction, is not necessary for the solution to hold. In $\mathfrak{D}_6$, axiom $\mathit{locIn}(l_2, \mathit{northeast}) \in \mathcal{D}_{-2}$ is simply redundant, as it is already present in the data stream. Prediction $\mathfrak{D}_3$ is $\preceq_s$-minimal and $\preceq_e$-minimal, yet not $\preceq_b$-minimal. Note that considering the background knowledge constraint $\mathit{SevereDroughtIn} \sqsubseteq \mathit{DraughtIn}$, axiom $\mathit{DraughtIn}(l_2)$ is logically weaker than the assumed $\mathit{SevereDroughtIn}(l_2) \in \mathcal{D}_2$, and could be possibly used to replace the latter in the solution. Finally, $\mathfrak{D}_4$ and $\mathfrak{D}_7$ are $\preceq_b$-minimal but not $\preceq_s$-minimal, as both can be turned into distinct solutions by subtracting the state $\mathcal{D}_2$ from the former and either of the empty states $\mathcal{D}_{-3}$ or $\mathcal{D}_{-2}$ from the latter.

5 Complexity of reasoning

In this section, we study the combined complexity of reasoning problems comprising different variants of prediction and explanation tasks. The proofs are included in the appendix. Note that a "recognition" result with respect to a minimality criterion signals that the underlying decision procedure is complete but not necessarily sound, i.e. the identified solutions might require an additional check for being minimal in the given sense. A "computation" result implies soundness as well [18].

We start by considering ABox abduction, i.e., the task of abducing a minimal ABox ensuring entailment and non-entailment of selected CQs, which is later generalized to sequences of such problems.

Definition 9 (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 grounded 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 8, 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 [18].

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 $\mathfrak{D}$ is fixed up to renaming individuals in the included ABoxes.

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 solving A-sequence abduction problems in TQL. Since technically abduction for future-present formulas is symmetric to abduction for past-present formulas, we only study the former setting, noting that all results transfer automatically to the latter. 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 10 (A-structures). *An A-structure is a tuple $\mathfrak{S} = (S, \mathcal{S}_0, \rightarrow)$, where S is a finite set of ABoxes, $\mathcal{S}_0 \in S$ is the initial ABox, and $\rightarrow : S \mapsto S$ is a transition function. The* unfolding *of $\mathfrak{S}$ is an A-sequence $\mathcal{S}_0, \ldots, \mathcal{S}_i, \mathcal{S}_{i+1}, \ldots$, where for every $i \in \mathbb{N}$, $\mathcal{S}_i \in S$ and $\mathcal{S}_i \rightarrow \mathcal{S}_{i+1}$.*

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 $\boldsymbol{s} = (s_i)_{i \in \mathbb{N}}$ is an abstraction of an infinite sequence of temporal states entailing a given A-sequence. Each s_i-th element $(t_i, \mathcal{A}(t_i))$ in that sequence consists of the set t_i of subformulas of ϕ that must be entailed in i and the minimal ABox $\mathcal{A}(t_i)$ that must hold at i for ϕ to be true at time 0. Particularly instrumental are special quasimodels called ultimately periodic, which consist 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 an A-sequence abduction 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 head of the quasimodel.

For A-sequence abduction over full TQL formulas the relevant A-structures are 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 [12]. 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 an A-sequence abduction 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 $\mathfrak{S} = (S, \mathcal{S}_0, \to)$ 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 A-sequence abduction problems is an adaptation of Sistla and Clarke's decision procedure for LTL [12]. 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 A-sequence solutions). *Recognizing an $\preceq_e$- and $\preceq_s$-minimal solution to an A-sequence abduction problem $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$, where ϕ is a* TQL *formula, is* PSpace*-complete.*

In case of $\text{TQL}^{\exists}$ 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 $\text{TQL}^{\exists}, \text{TQL}^{+}$). *Let $\mathfrak{D}$ be an $\preceq_e$- and $\preceq_s$-minimal solution to an A-sequence abduction problem $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$, where $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ and ϕ is a $\text{TQL}^{\exists}$ or TQL^{+} formula. Then there exists an A-structure $\mathfrak{S} = (S, \mathcal{S}_0, \to)$ 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 A-sequence solutions for $\text{TQL}^{\exists}$ drops to DB. 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 DB-complete.

Theorem 2 (Recognizing A-sequence solutions for $\text{TQL}^{\exists}$). *Recognizing an $\preceq_e$- and $\preceq_s$-minimal solution to an A-sequence abduction problem $(\mathcal{T}, \mathfrak{A}, \phi)$, where ϕ is a $\text{TQL}^{\exists}$ formula, is* DB*-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 $\text{TQL}^{\exists,+}$ does not yield any additional gain, even when $\preceq_b$-minimality is considered. This is a consequence of the non-determinism involved in choosing the order in which U-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.

Lemma 4 (Recognizing A-sequence solutions for $\mathrm{TQL}^{+}, \mathrm{TQL}^{\exists,+}$**).** *Recognizing a $\preceq_e$- and $\preceq_s$-minimal solution to an A-sequence abduction problem $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$, where ϕ is a TQL^{+} or $\mathrm{TQL}^{\exists,+}$ formula, is* NP*-complete. The result holds even for $\preceq_b$-minimal solutions and when $\mathfrak{A} = \emptyset$.*

Note that in most cases computing A-sequence 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.

As the last task considered in this section, we address entailment of TQL formulas by finite A-sequences. As explained in Section 4, this problem corresponds to deciding whether the antecedent of a temporal rule, in case of prediction, or its consequent, in explanation, is entailed by a given fragment of the recorded segment. In the following theorem, we show that the problem is DP-complete in general or NP-complete in a special case, where the difference is determined by the presence of lack of negative CQ occurrences.

Theorem 3 (Entailment by finite A-sequences). *Let $\mathcal{T}$ be a TBox and $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ an A-sequence, where I is a finite interval over $\mathbb{Z}$. Deciding $\mathcal{T}, \mathfrak{A}, n \models \phi$, for some $n \in \mathbb{Z}$, is DP-complete* iff *ϕ is a grounded* TQL *or* $\mathrm{TQL}^{\exists}$ *formula, and NP-complete* iff *ϕ is a grounded* $\mathrm{TQL}^{\exists,+}$ *formula.*

The analysis above shows that prediction and explanation 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. Less assumptive predictions and explanations (such as based on the $\preceq_b$-minimality criterion) and a simpler language for learning temporal association rules might moreover offer conjectures of a higher likelihood, thus offering another reward for the lost expressiveness.

6 Related work

To the best of our knowledge, prediction and explanation in the conceptual and technical sense considered here have not been addressed in the literature. Lecue and Pan study prediction over ontology streams in [3], but clearly follow the data mining approach to the problem, focusing on detection of statistical correlations in data and their future projections. Such a perspective is orthogonal to ours, as here we deal exclusively with the knowledge representation and reasoning level, assuming that relevant association rules are already given and symbolically expressed as temporal rules. In the report [5], Thirunarayan *et al.* propose to use abductive logic programming for generating explanations, understood as abstractions of quantitative data into qualitative descriptions, as an integral component of a situation awareness framework over the Semantic Sensor Web. Although the preliminary nature of this proposal does not allow for a detailed comparison with ours, it clearly follows a similar motivation and formal direction.

Other types of reasoning services over semantic streaming data, not of immediate relevance to this work, have been considered in a number of papers, e.g., [1,11,2]. Yet more remotely related work deals with prediction and temporal association rule mining in the field of relational databases [6,14], aspects of abductive reasoning in temporal logics [19], logics for causal reasoning [20], and prediction and explanation in other AI contexts [4].

7 Conclusions and outlook

In this paper, we have introduced a novel formalization of predictive and explanatory reasoning over DL-*Lite* data streams, and delivered a number of results characterizing the computational complexity of both tasks using different variants of the underlying temporal rule formalism. We believe that the approach we propose, which allows for studying prediction and explanation from the purely logical and computational perspective, is vital for the development of robust stream reasoning techniques applicable to semantically rich data, as it introduces a symbolic layer which can usefully mediate between the semantic and statistical view on the data.

An especially promising direction of advancing this work further is to investigate the use of other temporal logics for expressing temporal rules, in particular those offering real-time and probabilistic features, e.g., PCTL [21]. Arguably, such rules could be tighter aligned with typical models of causal reasoning [20] and the practice of temporal association rule learning [14]. As an alternative to the probabilistic approach, a qualitative one, based on defeasible semantics [22], could be also potentially useful. Considering the implementation prospects, a natural and technically feasible approach is likely to be found in combining temporal databases, recently supported via SQL:2011 [23], with existing reasoning tools enabling execution of temporal logic programs, such as MetateM [24].

References

1. Valle, E.D., Ceri, S., van Harmelen, F., Fensel, D.: It's a streaming world! reasoning upon rapidly changing information. IEEE Intelligent Systems **24**(6) (2009)

2. Barbieri, D.F., Braga, D., Ceri, S., Valle, E.D., Huang, Y., Tresp, V., Rettinger, A., Wermser, H.: Deductive and inductive stream reasoning for semantic social media analytics. IEEE Intelligent Systems **25**(6) (2010) 32–41
3. Lecue, F., Pan., J.Z.: Predicting Knowledge in An Ontology Stream. In: Proc. of the International Joint Conference on Artificial Intelligence (IJCAI-13). (2013)
4. Shanahan, M.: Prediction is deduction but explanation is abduction. In: Proc. of the International Joint Conference on Artificial Intelligence (IJCAI-89). (1989)
5. Thirunarayan, K., Henson, C.A., Sheth, A.P.: Situation awareness via abductive reasoning from semantic sensor data: A preliminary report. In: Proc. of the International Symposium on Collaborative Technologies and Systems (CTS-09). (2009)
6. Akdere, M., Çetintemel, U., Upfal, E.: Database-support for continuous prediction queries over streaming data. Proc. of the International Conference on Very Large Data Bases (VLDB-10) (2010)
7. 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) (2007) 385–429
8. 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)
9. 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)
10. 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)
11. Ren, Y., Pan, J.Z.: Optimising ontology stream reasoning with truth maintenance system. In: Proc. of the ACM Conference on Information and Knowledge Management (CIKM-11). (2011)
12. Sistla, A.P., Clarke, E.M.: The complexity of propositional linear temporal logics. Journal of ACM **32**(3) (1985) 733–749
13. 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)
14. Guillame-Bert, M., Crowley, J.L.: Learning temporal association rules on symbolic time sequences. Journal of Machine Learning Research - Proceedings Track **25** (2012) 159–174
15. Dhanya, C.T., Kumar, D.N.: Data mining for evolution of association rules for droughts and floods in India using climate inputs. Journal of Geophysical Research **114** (2009)
16. 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)
17. Klarman, S., Endriss, U., Schlobach, S.: Abox abduction in the description logic $\mathcal{ALC}$. Journal of Automated Reasoning **46** (2011) 43–80
18. 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)
19. Ribeiro, C., Porto, A.: Abduction in temporal reasoning. In: Proc. of the International Conference on Temporal Logic (ICTL-94). (1994)
20. Kleinberg, S., Mishra, B.: The temporal logic of causal structures. In: Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI-09)). (2009)

21. Hansson, H., Jonsson, B.: A logic for reasoning about time and reliability. Formal Aspects of Computing **6** (1994) 102–111
22. Governatori, G., Terenziani, P.: Temporal extensions to defeasible logic. In: Proc. of the Australian Conference on Advances in Artificial Intelligence. (2007)
23. Kulkarni, K., Michels, J.E.: Temporal features in SQL:2011. SIGMOD Rec. **41**(3) (2012)
24. Barringer, H., Fisher, M., Gabbay, D., Gough, G., Owens, R.: MetateM: An introduction. Formal Aspects of Computing **7**(5) (1995) 533–549

Appendix

Below we present full proofs of the results included in Section 5.

A.1 ABox abduction

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 $\mathfrak{D}$ is fixed up to renaming individuals in the included ABoxes.

Proof. (1) To compute a solution, up to renaming individual names, it suffices to ground the conjuncts of every $q \in P$, replacing the existentially bounded variables with fresh constants. If the resulting ABox $\mathcal{D}$ is consistent with $\mathcal{T}$ — a condition verifiable in time polynomial in the size of $\mathcal{T}, P$ — then $\mathcal{D}$ is the unique $\preceq_b$-minimal solution and the unique $\preceq_e$-minimal solution whenever $\mathcal{T} = \emptyset$. For the former conclusion, observe that grounding any other CQ than q in $q^{\mathcal{T}}$ for any $q \in P$ must result in a non-$\preceq_b$-minimal solution, while for the latter, that $q^{\mathcal{T}} = q$ for every $q \in P$, and so grounding q is the only way to ensure entailment $db(\mathcal{D}) \models q$. Note also, that grounding distinct variables with the same constant is always redundant, given the restriction of identifying $\mathfrak{D}$ up to renaming of constants. E.g., for $q = \exists x, y.(C(x) \wedge D(y))$, the grounding $\{C(a), D(a)\}$ is redundant as it can be obtained from $\{C(a), D(b)\}$ by renaming $b \mapsto a$, but not vice versa.

(2) The upper bound transfers from the case of $\mathcal{T} \neq \emptyset$ and $\mathcal{A} \neq \emptyset$, proved in [18] as one type of the recognition problems for negative query explanations. Note that the number of distinct $\preceq_e$-minimal solutions must be bounded by $\ell(\mathcal{T})^{\ell(P)} \cdot \ell(\mathcal{A})^{\ell(P)}$, where the first factor is the number of CQs in the FO rewriting of a CQ, and the second one is the number of possible groundings of a CQ, and so it is at most exponential in the size of the input. The hardness for $\mathcal{T} \neq \emptyset$ can be shown by reduction from the 3-SAT problem. Let $f = c_1 \wedge \ldots \wedge c_n$ be a formula in CNF, where each $c_i = L_{i1} \vee L_{i2} \vee L_{i3}$ and every L_{ik} is a literal. We fix CQ $q = \exists x.(C_1(x) \wedge \ldots \wedge C_n(x))$, where C_i is a fresh concept name associated with the clause c_i, and define TBox encoding the clauses $\{L_{i1} \sqsubseteq C_i, L_{i2} \sqsubseteq C_i, L_{i3} \sqsubseteq C_i\}$ and the disjontness axioms for the complementary literals $L_p \sqsubseteq \neg \overline{L_p}$ where L_p is a concept name associated with atom p and $\overline{L_p}$ with $\neg p$. Then the formula f is satisfiable *iff* there exists a solution to the problem $(\mathcal{T}, \emptyset, \{q\}, \emptyset)$ in which only concepts $L_p, \overline{L_p}$ occur. Note, that the latter condition can be verified in time linear in $\mathcal{D}$, and so it does not add to the complexity of the problem. The hardness for $\mathcal{A} \neq \emptyset$ can be shown by reduction from the graph homomorphism problem. Given graphs $G = (V, E), G' = (V', E')$ we want to decide whether there exists a function $h : V \mapsto V'$ such that $(v, u) \in E$ implies $(h(v), h(u)) \in E'$. We encode

graph G' as the ABox $\mathcal{A}$, using a single role *edge* and unique individual names representing vertices, and G as the query using the same role and existentially bounded variables for the vertices. Then a requested homomorphism exists *iff* $\mathcal{D} = \emptyset$ is recognized as a $\preceq_e$-minimal solution.

(3) Observe that whenever $\mathcal{D}$ is an $\preceq_e$-minimal solution to $(\mathcal{T}, \mathcal{A}, P, N)$ for $N = \emptyset$, then for any $N \neq \emptyset$ it must be either still a $\preceq_e$-minimal solution or it is not a solution at all. The DP algorithm for an arbitrary problem $(\mathcal{T}, \mathcal{A}, P, N)$ first generates a candidate solution $\mathcal{D}$ by means of the NP algorithm used in (2), and then ensures it is a minimal one (in either sense $\preceq_e$ or $\preceq_b$) by executing a coNP procedure which attempts to find an alternative solution $(D)'$ refuting the minimality of $\mathcal{D}$. Finally, it checks that for every $q \in N$ it is the case that $\mathcal{T}, \mathcal{A} \cup \mathcal{D} \not\models q$. The latter problem is clearly coNP-complete, considering NP-completeness of CQ answering in the considered DL-*Lite* languages. Naturally, this holds even when $\mathcal{A} = \emptyset$. For hardness we consider any language $L \in \text{NP} \cap \text{coNP}$, i.e., such that $L = L_1 \cap L_2$ with $L_1 \in \text{NP}$ and $L_2 \in \text{coNP}$. Naturally, for any input x, it must be that $x \in L$ *iff* $x \in L_1$ and $x \in L_2$. But then there must exist a pair of polynomial reductions R_1, R_2 from L_1 and L_2 to some instances of CQ entailment and non-entailment problems. Note that by involving suitable vocabulary renaming, both target problems can use the same $\mathcal{D}$ and $\mathcal{T}$. Hence, finding an ABox requested in the lemma must be at least as hard as deciding $x \in L$. ❑

A.2 Types, state types, quasimodels

To simplify the proofs of the remaining results, without loss of generality we assume all A-sequence abduction problems to be fixed at time 0, i.e., such that for $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$ with $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ it is the case that $I = [0, I^+]$ for some $I^+ \in \mathbb{N}$. The solutions to such problems are consequently A-sequences $\mathfrak{D} = (\mathcal{D}_i)_{i \in \mathbb{N}}$. Further, we introduce some auxiliary nomenclature. Consider an A-sequence abduction problem $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$. Let $\mathsf{sub}(\phi)$ denote the set of all subformulas of ϕ and their complements. We assume that all occurrences of double negation symbols in $\mathsf{sub}(\phi)$ are removed and we write $\neg\psi$ to refer to the complement of formula $\psi \in \mathsf{sub}(\phi)$. A *type* for ϕ is a set $t \subseteq \mathsf{sub}(\phi)$ such that:

- $\psi \wedge \varphi \in t$ *iff* $\{\psi, \varphi\} \subseteq t$, for every $\psi \wedge \varphi \in \mathsf{sub}(\phi)$,
- $\psi \in t$ *iff* $\neg\psi \notin t$, for every $\psi \in \mathsf{sub}(\phi)$.

By T we denote the set of all types for ϕ. Clearly, $|\mathsf{sub}(\phi)| \leq 4\ell(\phi)$ and so $|T| \leq 2^{4\ell(\phi)}$. A *state type* is a pair $s = (t, \mathcal{A}(t))$, where $t \in T$ and $\mathcal{A}(t)$ is an ABox. A *quasimodel* for Ω is a sequence of state types $\boldsymbol{s} = (s_i)_{i \in \mathbb{N}}$, such that for $s_i = (t_i, \mathcal{A}(t_i))$, with $i \in \mathbb{N}$ specified as below, it holds that:

- $\phi \in t_i$, for $i = 0$,
- $\varphi \mathsf{U} \psi \in t_i$ *iff* there exists $j > i$ such that $\psi \in t_j$ and $\varphi \in t_k$ for every $i < k < j$, for every $\varphi \mathsf{U} \psi \in \mathsf{sub}(\phi)$ and $i \in \mathbb{N}$,
- $\mathcal{A}(t_i)$ is a $\preceq_e$-minimal solution to $(\mathcal{T}, \mathcal{A}_i, P, N)$, where $P = \{q \mid [q] \in t_i\}$ and $N = \{q \mid \neg[q] \in t_i\}$, for every $i \leq I^+$,

– $\mathcal{A}(t_i)$ is a $\preceq_e$-minimal solution to $(\mathcal{T}, \emptyset, P, N)$, where $P = \{q \mid [q] \in t_i\}$ and $N = \{q \mid \neg[q] \in t_i\}$, for every $i > I^+$.

A quasimodel $\boldsymbol{s} = (s_i)_{i \in \mathbb{N}}$ is called *ultimately periodic*, with the head of length $l > 0$ and the period $n \in \{1, \ldots, l\}$, *iff* $s_{i+kn} = s_i$, for every $i \geq l - n$ and $k \in J$ (cf. Figure 2).

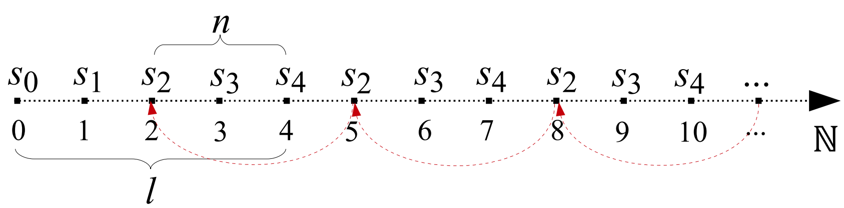

Fig. 2. An ultimately periodic quasimodel $\boldsymbol{s}$, with $l = 5$ and $n = 3$.

The following is a crucial property relating the structure of quasimodels with the semantics of TQL.

Proposition 1. *Let $\boldsymbol{s} = (s_i)_{i \in \mathbb{N}}$ be a quasimodel for Ω with $s_i = s_j$, for some $I^+ \leq i < j$. Let further $\boldsymbol{s}' = s_{h(0)}, \ldots, s_{h(i)}, s_{h(j+1)}, \ldots$ be a sequence of state types obtained from $\boldsymbol{s}$ by removing the subsequence $s_{i+1}, \ldots, s_j$ and revising the indexing of the remaining state types via a mapping $h : \{0, \ldots, i, j+1, \ldots\} \mapsto \mathbb{N}$, such that $h(k) = k$, for every $k \leq i$, and $h(k) = k - (j - i)$, for every $k \geq j + 1$. Then $\boldsymbol{s}'$ is a quasimodel for Ω.*

The argument builds on the observation that i entails exactly the same subformulas of ϕ as j. Moreover, $\mathcal{A}(t_l)$, for any $l > i \leq I^+$, depends exclusively on t_i. Hence, by structural induction over TQL, it follows that no formula in t_i can distinguish between sequences $s_{i+1}, s_{i+2}, \ldots$ and $s_{j+1}, s_{j+2}, \ldots$. Consequently, ϕ cannot distinguish between $\boldsymbol{s}$ and $\boldsymbol{s}'$ at time 0.

Every quasimodel $\boldsymbol{s}$ for Ω can be uniquely associated with a $\preceq_e$-minimal solution $\mathfrak{D}$ to Ω, namely the one constructed by fixing $\mathcal{D}_i = \mathcal{A}(t_i)$, for every $i \in \mathbb{N}$, $s_i = (t_i, \mathcal{A}(t_i))$. Conversely, every $\preceq_e$-minimal solution to Ω determines uniquely the corresponding quasimodel, considering that the choice of the ABox $\mathcal{A}(t_i)$, for every t_i, unambiguously determines entailment of subformulas $[q]$ and $\neg[q]$ in t_i, for every $q \in \mathsf{sub}(\phi)$, which in turn, by structural induction over ϕ, uniquely determine entailment of every subformula $\psi \in \mathsf{sub}(\phi)$ in t_i. Consequently, we note the following fact.

Proposition 2. *Let $\mathfrak{D}, \mathfrak{D}'$ be two $\preceq_e$-minimal solutions to $(\mathcal{T}, \mathfrak{A}, \phi)$, and $\boldsymbol{s}, \boldsymbol{s}'$ the quasimodels for Ω, associated with $\mathfrak{D}$ and $\mathfrak{D}'$, respectively. Then $\mathfrak{D} = \mathfrak{D}'$* iff *$\boldsymbol{s} = \boldsymbol{s}'$.*

A.3 A-sequence abduction in TQL

Lemma 2 (A-sequence vs. A-structure). *Let $\mathfrak{D}$ be an $\preceq_e$- and $\preceq_s$-minimal solution to an A-sequence abduction 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 $\mathfrak{S} = (S, \mathcal{S}_0, \to)$ whose unfolding is $\mathfrak{D}$, such that $|S| = f(\ell(\Omega))$, for some function $f(x) \in O(2^x)$.*

Proof. We claim that since $\mathfrak{D}$ is $\preceq_s$-minimal then the quasimodel for Ω associated with $\mathfrak{D}$ must be ultimately periodic (see Section 5 for the definition of ultimately periodic quasimodels), with the head of length $l \leq |T| + I^+$. Suppose $\boldsymbol{s}$ is a quasimodel associated with $\mathfrak{D}$, where $s_i = s_j$ for some $I^+ \leq i < j$. By Proposition 1, we can construct an alternative quasimodel $\boldsymbol{s}' = s_{h(0)}, \ldots, s_{h(i)}, s_{h(j+1)}, \ldots$. Then either it holds that $\boldsymbol{s}' \neq \boldsymbol{s}$ or $\boldsymbol{s}' = \boldsymbol{s}$. Suppose the first case applies. Then by Proposition 2, $\boldsymbol{s}'$ must be associated with some solution $\mathfrak{D}' \neq \mathfrak{D}$. Clearly, however, $\mathfrak{D}' \rightharpoonup \mathfrak{D}$ (where h is the mapping warranting the relation $\rightharpoonup$), and so $\mathfrak{D}$ is not $\preceq_s$-minimal, which contradicts the assumption. Alternatively, consider the latter situation. Then it follows that sequence $s_{i+1}, \ldots, s_j$ belongs to the periodic fragment of $\boldsymbol{s}$, where $kn = j - i$ for the period n and some $k \in \mathbb{N}$. This conclusion follows by induction over the structure of $\boldsymbol{s}$. Observe that the sequence $s_{j+1}, \ldots, s_{j+1+(j-i)}$ in $\boldsymbol{s}$ must be equal to $s_{i+1}, \ldots, s_j$ or else it would not be the case that $\boldsymbol{s} = \boldsymbol{s}'$. But then, by the same token, the follow-up sequence of the same length must be equal to $s_{j+1}, \ldots, s_{j+1+(j-i)}$, and so on. Finally, consider some $s_i = (t_i, \mathcal{A}(t_i))$ and $s_j = (t_j, \mathcal{A}(t_j))$ in $\boldsymbol{s}$, such that $I^+ \leq i < j$, $t_i = t_j$ and $\mathcal{A}(t_i) \neq \mathcal{A}(t_j)$. Then by fixing $s_j := (t_j, \mathcal{A}(t_i))$ we obtain an alternative quasimodel $\boldsymbol{s}'$ in which $s_i = s_j$, and the entire argument above applies again. Clearly, there must exist a fixpoint at which any further application of the argument from Proposition 1 returns consistently the same (ultimately periodic) quasimodel. At that point the head of that quasimodel consists of at most I^+ initial state types, corresponding to $\mathfrak{A}$, followed by at most $|T|$ unique state types. No later than at that point the first duplicate state type in $\boldsymbol{s}$ must occur, marking the end of the first period in the quasimodel.

Given the existence of the quasimodel $\boldsymbol{s}$ for Ω, with above stated properties the construction of an A-structure $\mathfrak{S} = (S, \mathcal{S}_0, \to)$ postulated by the lemma is straightforward. We set $\mathcal{S}_i := s_i$ and $\mathcal{S}_i \to \mathcal{S}_{i+1}$ for every $i < l - 1$, and $\mathcal{S}_{l-1} := s_{l-1}$, $S_{l-1} \to \mathcal{S}_{l-n}$. By the construction of $\mathcal{S}$, definition of quasimodels and their ultimate periodicity, demonstrated above, it follows that $\mathfrak{D}$ must be the unfolding of $\mathfrak{S}$. Clearly, $|S| \leq |T| + I^+$, where I^+ is linear in $\ell(\Omega)$. Therefore, there exists a function $f(x) \in O(2^x)$, such that $|S| \leq f(\ell(\Omega))$. ❏

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

Proof. The hardness transfers from the satisfiability problem in LTL. Note that CQs in TQL can be used simply as propositions, where the CQ associated with proposition p is fixed as $q_p = A_p(x)$, for a designated concept name A_p. Then an LTL formula is satisfiable *iff* there exists a $\preceq_e$- and $\preceq_s$-minimal solution to the

A-sequence problem $(\emptyset, \emptyset, \phi)$, where ϕ is the corresponding TQL query, grounded with $x \mapsto a$, for some unique $a \in N_I$. Recall, that satisfiable LTL formulas must have ultimately periodic models of at most exponential size [12].

Next, we establish the upper bound by augmenting the decision procedure for LTL with an additional DP routine which computes solutions to the ABox abduction problems handled in the consecutive states of a generated quasimodel. At the start, the algorithm guesses two numbers: the length of the head $l \leq |T| + I^+$ and the period $n \in \{1, \ldots, l\}$. Then it non-deterministically picks a type t_0 for ϕ such that $\phi \in t_0$, and selects $\mathcal{A}(t_0)$). The latter choice is made using a DP routine described in Lemma 1, in such a way that the suitable conditions in the definition of the quasimodel are satisfied. Then for every $1 \leq i \leq l$, the algorithm picks a type t_i and $\mathcal{A}(t_i)$ and ensures the following conditions hold:

- for every $\varphi \mathsf{U} \psi \in t_{i-1}$, if $\neg\psi \in t_i$ then $\varphi \mathsf{U} \psi \in t_i$ and $\varphi \in t_i$,
- for every $\varphi \mathsf{U} \psi \in t_i$, if $\varphi \in t_i$ then $\varphi \mathsf{U} \psi \in t_{i-1}$,
- if $\psi \in t_i$ then $\varphi \mathsf{U} \psi \in t_{i-1}$, for every $\varphi \mathsf{U} \psi \in \mathsf{sub}(\phi)$,
- for every $\varphi \mathsf{U} \psi \in t_i$, there is j such that $\psi \in t_j$ and:
  - $i < j \leq l$, whenever $i < l - n$,
  - $l - n \leq j \leq l$, whenever $l - n \leq i \leq l$,
- $t_l = t_{l-n}$ and $\mathcal{A}(t_l) = \mathcal{A}(t_{l-n})$.

It is not difficult to observe that a sequence of state types satisfying the conditions above must be in fact an ultimately periodic quasimodel for Ω. During its run, the algorithm requires at most polynomial space of the working memory, in order to store three state types (t_{l-n} and the current pair t_i, t_{i+1}) and the set of U-formulas that have to be still fulfilled in the future. The generated sequence is systematically written on the output tape during the computation process and ended with a designated symbol marking that the sequence is eventually accepted by the procedure. By Lemma 2, every $\preceq_e$- and $\preceq_s$-minimal solution to Ω must be found as one of the outputs. We thus obtain a NPSPACE procedure, which by Savage's theorem is in PSPACE. ❏

A.4 A-sequence abduction in $\mathbf{TQL}^{\exists}, \mathbf{TQL}^{+}, \mathbf{TQL}^{\exists,+}$

Lemma 3 (A-sequence vs. A-structure for $\mathrm{TQL}^{\exists}, \mathrm{TQL}^{+}$**).** *Let $\mathfrak{D}$ be an $\preceq_e$- and $\preceq_s$-minimal solution to an A-sequence abduction problem $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$, where $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ and ϕ is a $\mathrm{TQL}^{\exists}$ or TQL^{+} formula. Then there exists an A-structure $\mathfrak{S} = (S, \mathcal{S}_0, \rightarrow)$ whose unfolding is $\mathfrak{D}$, such that $|S| \leq f(\ell(\phi))$, for some $f(x) \in O(x)$.*

Proof. As the starting point we consider the result in Lemma 2, and the argument used in its proof. Here, we essentially show that that argument can be pushed further in case of $\mathrm{TQL}^{\exists}$ and TQL^{+}, leading to a smaller upper bound on the size of relevant A-structures, with $|S| \leq 2\ell(\phi) + I^+$. Consider an $\preceq_e$- and $\preceq_s$-minimal solution $\mathfrak{D}$ and its corresponding, ultimately periodic quasimodel $\boldsymbol{s}$ with the head of length l and the period n. We show that $l \leq 2|\ell(\phi)| + I^+$ or else $\mathfrak{D}$ cannot be $\preceq_s$-minimal. Let $F = \{\varphi \mathsf{U} \psi \mid \varphi \mathsf{U} \psi \in \bigcup_{0 \leq i < l} t_i\}$, i.e., F is the set of U-formulas used in the head of $\boldsymbol{s}$, which is obviously equivalent to the set of U-formulas used in the entire quasimodel (note that $\varphi = \top$ in case of $\mathrm{TQL}^{\exists}$). Clearly, $|F| \leq \frac{|\mathsf{sub}(\phi)|}{2}$. By the semantics of TQL and the construction of the quasimodel, for every $\varphi \mathsf{U} \psi \in F$ and $0 \leq i < l - n$, whenever $\varphi \mathsf{U} \psi \in t_i$ then there must exist $i < j < l$, such that $\psi \in t_j$. Similarly, for $l - n \leq i < l$, if $\varphi \mathsf{U} \psi \in t_i$ then $\psi \in t_j$, for some $l - n \leq j < l$. For every $\varphi \mathsf{U} \psi \in F$ let $g(\psi)$ be the largest number $g(\psi) < l$ such that $\psi \in t_{g(\psi)}$. Finally, we mark selected state types in the head of $\boldsymbol{s}$ by running the following procedure until saturation:

- s_i is marked, for every $0 \leq i \leq I^+$,
- for every $0 \leq i < l$, if $s_i = (t_i, \mathcal{A}(t_i))$ is marked and $\varphi \mathsf{U} \psi \in t_i$, for some $\varphi \mathsf{U} \psi \in F$, then mark $s_{g(\psi)}$.

Clearly, there can be at most $|F| + I^+$ state types marked after the procedure terminates. Remove all state types that are not marked and consider the remaining sequence, with a suitable revised indexing. It is not difficult to see, that this sequence is in fact the head of an ultimately periodic quasimodel $\boldsymbol{s'}$ for Ω. We can thus follow an argument from the proof of Lemma 2 and consider two disjoint cases: $\boldsymbol{s} \neq \boldsymbol{s'}$ or $\boldsymbol{s} = \boldsymbol{s'}$. In the first scenario, we conclude that the solution $\mathfrak{D}$ cannot be in fact $\preceq_s$-minimal, which contradicts the original assumption. Hence the latter must be true. But this means that all state types in the head of $\boldsymbol{s}$ must have been marked by the procedure, and so the length of the head is bounded by $|F| + I^+$, i.e., $l \leq 2\ell(\phi) + I^+$. The final A-structure is constructed exactly as in the proof of Lemma 2. ❏

Theorem 2 (Recognizing A-sequence solutions for $\mathrm{TQL}^{\exists}$). *Recognizing an $\preceq_e$- and $\preceq_s$-minimal solution to an A-sequence abduction problem $(\mathcal{T}, \mathfrak{A}, \phi)$, where ϕ is a $\mathrm{TQL}^{\exists}$ formula, is* DB-*complete.*

Proof. For the upper bound we consider an algorithm, which first guesses the numbers $l \leq 2\ell(\phi) + I^+, n \in \{1, l\}$, and next it non-deterministically generates a sequence of types $t_0, \dots t_l$ alongside the corresponding ABoxes $\mathcal{A}(t_0), \dots, \mathcal{A}(t_l)$. The latter step involves a DP routine, as described in the proof of Lemma 1, requested to satisfy the criteria characterizing quasimodels. For every $0 \leq i \leq l$, the algorithm verifies satisfaction of the following conditions:

- $\top \mathsf{U} \psi \in t_{i-1}$, for every $\top \mathsf{U} \psi \in t_i$,
- if $\psi \in t_i$ then $\top \mathsf{U} \psi \in t_{i-1}$, for every $\top \mathsf{U} \psi \in \mathsf{sub}(\phi)$,
- for every $\top \mathsf{U} \psi \in t_i$, , there is j such that $\psi \in t_j$ and:
    - $i < j \leq l$, whenever $i < l - n$,
    - $l - n \leq j \leq l$, whenever $l - n \leq i \leq l$,
- $t_l = t_{l-n}$ and $\mathcal{A}(t_l) = \mathcal{A}(t_{l-n})$.

Whenever the conditions are satisfied, the sequence $\mathfrak{D}$, such that $\mathcal{D}_i = \mathcal{A}_i$, for every $0 \leq i < l$, is returned as a relevant solution to the problem Ω. By Lemma 3, every $\preceq_e$- and $\preceq_s$-minimal solution to Ω must be found as one of the outputs. The lower bound follows by reduction from an arbitrary DP-complete problem, conducted precisely as in the proof of Lemma 1, point 3, where the entailment and non-entailment of CQs are again the target NP- and co-NP-complete problems in the reduction. ❑

Lemma 4 (Recognizing A-sequence solutions for $\mathrm{TQL}^{+}, \mathrm{TQL}^{\exists,+}$**).** *Recognizing a* $\preceq_e$*- and* $\preceq_s$*-minimal solution to an A-sequence abduction problem* $\Omega = (\mathcal{T}, \mathfrak{A}, \phi)$*, where* ϕ *is a* TQL^{+} *or* $\mathrm{TQL}^{\exists,+}$ *formula, is* NP*-complete. The result holds even for* $\preceq_b$*-minimal solutions and when* $\mathfrak{A} = \emptyset$*.*

Proof. The upper bound follows by the algorithm analogical to that used in Theorem 2. The only difference is that given a type t for Ω a corresponding solution $\mathcal{A}(t)$ can be computed at worst in NP (by Lemma 1, points 1, 2). Hence the algorithm must only guess the suitable sequence of types, which is a problem in NP.

The lower bound is demonstrated by reduction from the NP-complete Hamiltonian path problem, defined as follows: given a directed graph $G = (V, E)$ decide whether there exists a path through G which visits every vertex exactly once. With every vertex $v \in V$, we associate a query $q_v = A_v(x)$, for a designated concept name A_v. Consider a formula $\phi = \bigwedge_{v \in V}(\top \mathsf{U} q_v)$ grounded with $x \mapsto a$, for some unique $a \in N_I$. Then there exists a Hamiltonian path through G *iff* there exists a $\preceq_{b/e}$- and $\preceq_s$-minimal solution $\mathfrak{D}$ to $(\emptyset, \emptyset, \phi)$ such that:

- $\mathcal{D}_0 = \emptyset$,
- there exists a bijection $h : V \mapsto \{1, \ldots, |V|\}$, such that for every $v \in V$:
  - $A_v(a) \in \mathcal{D}_{h(v)}$,
  - $A_u(a) \notin \mathcal{D}_{h(v)}$, for every $u \in V$ with $u \neq v$,
  - $(v, u) \in E$, for $u \in V$ such that $h(u) = h(v) + 1$.

Observe that for a given A-structure $\mathfrak{S}$, associated with $\mathfrak{D}$, verifying the conditions above can be done in time linear in the size of $\mathfrak{S}$, and thus in the size of the input. Hence, the verification step does not add to the complexity of the problem. Clearly, whenever $\mathfrak{D}$ does satisfy the conditions above it contains the hamiltonian path through G, given via h. Conversely, suppose that there exists a Hamiltonian path through G. Then clearly there must exist an A-sequence $\mathfrak{D}$, described as above, which solves $(\emptyset, \emptyset, \phi)$. It is not difficult to see that such an A-sequence is both $\preceq_{b/e}$- and $\preceq_s$-minimal. ❑

A.5 Entailment of $\mathbf{TQL}^{\exists}, \mathbf{TQL}^{+}, \mathbf{TQL}^{\exists,+}$ formulas

Theorem 3 (Entailment by finite A-sequences). *Let* $\mathcal{T}$ *be a TBox and* $\mathfrak{A} = (\mathcal{A}_i)_{i \in I}$ *an A-sequence, where* I *is a finite interval over* $\mathbb{Z}$*. Deciding* $\mathcal{T}, \mathfrak{A}, n \models \phi$*, for some* $n \in \mathbb{Z}$*, is DP-complete* iff ϕ *is a grounded* TQL *or* $\mathrm{TQL}^{\exists}$ *formula, and NP-complete* iff ϕ *is a grounded* $\mathrm{TQL}^{\exists,+}$ *formula.*

Proof. Again, w.l.o.g. we assume that $I^- = n = 0$. The upper bound follows by an algorithm which first non-deterministically selects a sequence of types $t_0, \ldots, t_n$ for ϕ, and next for every $0 \leq i \leq I^+$ it decides whether $\mathcal{T}, \mathfrak{A}, 0 \models [q]$, for every $[q] \in t_i$, and $\mathcal{T}, \mathfrak{A}, 0 \models \neg[q]$, for every $\neg[q] \in t_i$. If so, then ϕ is entailed. The two decision problems are in NP and co-NP, respectively. In case of $\text{TQL}^{\exists,+}$ only the first type of problems has to be addressed. The algorithm runs therefore in DP for TQL and $\text{TQL}^{\exists}$, and in NP for $\text{TQL}^{\exists,+}$. The lower bound follows again by reduction from an arbitrary DP-complete problem (see Lemma 1, point 3) for TQL and $\text{TQL}^{\exists}$. For $\text{TQL}^{\exists,+}$ the lower bound transfers from the CQ entailment problem. ❑