Temporalized $\mathcal{EL}$ Ontologies for Accessing Temporal Data: Complexity of Atomic Queries

Víctor Gutiérrez-Basulto[1] and **Jean Christoph Jung**[1] and **Roman Kontchakov**[2]
[1] Department of Computer Science, Universität Bremen, Germany
[2] Department of Computer Science and Information Systems, Birkbeck, University of London, UK

Abstract

We study access to temporal data with $\mathcal{TEL}$, a temporal extension of the tractable description logic $\mathcal{EL}$. Our aim is to establish a clear computational complexity landscape for the atomic query answering problem, in terms of both data and combined complexity. Atomic queries in full $\mathcal{TEL}$ turn out to be undecidable even in data complexity. Motivated by the negative result, we identify well-behaved yet expressive fragments of $\mathcal{TEL}$. Our main contributions are a semantic and sufficient syntactic conditions for decidability and three orthogonal tractable fragments, which are based on restricted use of rigid roles, temporal operators, and novel acyclicity conditions on the ontologies.

1 Introduction

In recent years, the use of ontologies to enrich plain data with a semantic layer has become one of the outstanding applications of description logic (DLs) technologies in the Semantic Web. The ontology-based data access (OBDA) setting provides information systems with various advantages, e.g., a friendlier vocabulary for accessing heterogenous data is given by the ontology, and means of querying potentially incomplete data are provided by taking account of the implicit knowledge derived from the data and the ontology. Due to the increasing need to account for the temporal dimension of data available on the Web [Roth and Tan, 2013; Dong and Tan, 2015], the DL community has recently investigated extensions of the OBDA paradigm for temporal data. The initial efforts concentrated on temporal query languages with atemporal ontologies [Gutiérrez-Basulto and Klarman, 2012; Klarman and Meyer, 2014; Baader, Borgwardt, and Lippmann, 2015; Borgwardt, Lippmann, and Thost, 2015; Borgwardt and Thost, 2015]. On the other hand, temporal ontology languages can enhance conceptual modelling with temporal aspects [Artale et al., 2015], which are required, e.g., in applications managing data from sensor networks. In this line, the research has focused on temporal extensions of *DL-Lite* that support rewritability of temporal queries into the monadic second-order logic with order or into two-sorted first-order logic with $<$ and $+$ [Artale et al., 2013b; 2015]. Since standard relational database management systems have such built-in predicates, they can in principle evaluate the $\mathrm{FO}(<,+)$-rewritings. However, no temporal extensions of other classical DLs have been investigated yet in the context of OBDA, which is partly because of the intractability and often even undecidability of the standard reasoning tasks (e.g., subsumption) [Artale et al., 2007; Gutiérrez-Basulto, Jung, and Lutz, 2012; Gutiérrez-Basulto, Jung, and Schneider, 2014]. On the other hand, temporal data has also been studied in classical database theory [Chomicki and Toman, 2005]. In their seminal paper, Chomicki and Imielinski [1988] identified $\textsc{Datalog}_{1S}$ as a decidable extension of $\textsc{Datalog}$ with one successor function. Here we make the first (to the best of our knowledge) attempt to link temporal OBDA with temporal deductive databases [Chomicki, 1990; Baudinet, Chomicki, and Wolper, 1993].

In this paper, we study $\mathcal{TEL}$, a temporal extension of $\mathcal{EL}$ [Baader, Brandt, and Lutz, 2005]. The underlying DL component, $\mathcal{EL}$, underpins the OWL 2 EL profile of OWL 2 and the medical ontology SNOMED CT, which provides the vocabulary for electronic health records (EHRs). Indeed, applications managing EHRs must be able to provide information, e.g., on when and for how long some drug has been prescribed to a patient, so that drugs that interact adversely are not prescribed at the same time. Clinical trials [Shankar et al., 2008; O'Connor et al., 2009] also require a unified conceptual model for specifying temporal constraints of protocol entities such as *'a viable participant should have had a vaccination with live virus 5 days ago'* or '*blood tests of a patient should be run every 3 days*'. These statements can be encoded in $\mathcal{TEL}$:

$$\mathsf{Patient} \sqcap \bigcirc_P^5 \exists \mathsf{vaccinated}.\mathsf{LiveVirus} \sqsubseteq \mathsf{ViableParticip}, \quad (1)$$

$$\mathsf{Patient} \sqcap \bigcirc_P^3 \mathsf{ReqBloodTest} \sqsubseteq \mathsf{ReqBloodTest}. \quad (2)$$

Our main objective is to establish the limits of decidability and tractability of the query answering problem over $\mathcal{TEL}$ ontologies, in terms of both data and combined complexity. In order to set the foundations, we focus on *temporal atomic queries*. On the one hand, an atomic query $\mathsf{ViableParticip}(x,t)$ with the temporal concept inclusion (1) effectively encodes a tree-shaped temporal conjunctive query. On the other hand, using (1) to extend the vocabulary with a concept $\mathsf{ViableParticip}$ is closer to the spirit of the OBDA paradigm than repeating the same conjunction in all similar user queries. Moreover, a recurrent pattern $\mathsf{ReqBloodTest}$ is expressible as an atomic query

$\mathsf{ReqBloodTest}(x, t)$ with the temporal concept inclusion (2) but not expressible as a query without temporal concept inclusions like (2). As we shall see, even for atomic queries rather surprising (and challenging) results are obtained.

Our main contributions are complexity bounds, algorithms and rewritability into DATALOG_{1S} for atomic query answering in fragments of $\mathcal{TEL}$. Since query answering over unrestricted $\mathcal{TEL}$ turns out to be undecidable (in data complexity), we investigate its fragments to attain decidability and tractability. First, for $\mathcal{TEL}^{\bigcirc}$, which allows only the 'next-' $\bigcirc_F$ and 'previous-time' $\bigcirc_P$ operators, we identify *ultimate periodicity* as a natural semantic condition ensuring decidability, more precisely, PSPACE data complexity (the question of decidability of the full $\mathcal{TEL}^{\bigcirc}$ is left open for future work). Then, we identify a number of fragments with better computational properties. (*i*) For the fragment of $\mathcal{TEL}^{\bigcirc}$ without rigid (not changing over time) roles on the right-hand side of concept inclusions, we construct a polynomial rewriting into DATALOG_{1S}, and so, establish PSPACE-completeness for data complexity. This fragment contains all $\mathcal{EL}$ ontologies as well as both (1) and (2). (*ii*) Over *temporally acyclic* $\mathcal{TEL}^{\bigcirc}$-ontologies (with rigid roles and concepts), query answering is PTIME-complete in both data and combined complexity. This tractable fragment contains (1) and fully captures all atemporal $\mathcal{EL}$ ontologies and may prove particularly useful in applications; it, however, does not contain (2). (*iii*) Query answering over *DL-acyclic* $\mathcal{TEL}^{\bigcirc}$ ontologies is NC^1-complete for data complexity (in principle, highly parallelizable). This fragment contains many acyclic $\mathcal{EL}$ ontologies as well as both (1) and (2) (note that large parts of SNOMED CT are in fact acyclic). We remark that our two novel acyclicity conditions (each constraining only one dimension) are inspired by the 'traditional' notion of acyclicity in (temporal extensions of) DLs [Haase and Lutz, 2008; Gutiérrez-Basulto, Jung, and Schneider, 2015]. Finally, (*iv*) we show that the language with only $\Diamond_P$ and $\Diamond_F$ (sometime in the past/future) on the left-hand side of concept inclusions also enjoys PTIME query answering.

2 Preliminaries

We begin by introducing $\mathcal{TEL}$, a temporal extension of the classical DL $\mathcal{EL}$. Let $\mathsf{N_C}$, $\mathsf{N_R}$, $\mathsf{N_I}$ be countably infinite sets of *concept*, *role* and *individual names*, respectively. We assume that $\mathsf{N_R}$ is partitioned into two infinite sets, $\mathsf{N_R^{rig}}$ and $\mathsf{N_R^{loc}}$, of *rigid* and *local role names*, respectively. $\mathcal{TEL}$ *concepts* are defined by the following grammar:

$$C, D \ ::= \ A \ \mid \ C \sqcap D \ \mid \ \exists r.C \ \mid \ \bigcirc_* C \ \mid \ \Diamond_* C,$$

where $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $* \in \{F, P\}$. A $\mathcal{TEL}$*-TBox* (*ontology*) $\mathcal{T}$ is a finite set of *concept inclusions* (*CIs*) $C \sqsubseteq D$ and *concept definitions* (*CDs*) $C \equiv D$ for $\mathcal{TEL}$ concepts C, D. Data is given in terms of *temporal ABoxes* $\mathcal{A}$, which are finite sets of assertions of the form $A(a, n)$ and $r(a, b, n)$, where $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, $a, b \in \mathsf{N_I}$, and $n \in \mathbb{Z}$. We denote by $\mathsf{ind}(\mathcal{A})$ the sets of individual names occurring in $\mathcal{A}$, and by $\mathsf{tem}(\mathcal{A})$ the set $\{n \in \mathbb{Z} \mid \min \mathcal{A} \leq n \leq \max \mathcal{A}\}$, where $\min \mathcal{A}$ and $\max \mathcal{A}$ are, respectively, the minimal and maximal time points in $\mathcal{A}$. The size, $|\mathcal{T}|$ and $|\mathcal{A}|$, of $\mathcal{T}$ and $\mathcal{A}$ is the number of symbols required to write $\mathcal{T}$ and $\mathcal{A}$, respectively, with time points $n \in \mathbb{Z}$ encoded in *unary*. A *temporal knowledge base* (*KB*) $\mathcal{K}$ is a pair $(\mathcal{T}, \mathcal{A})$.

An *interpretation* $\mathfrak{I}$ is a structure $(\Delta^{\mathfrak{I}}, (\mathcal{I}_n)_{n \in \mathbb{Z}})$, where each $\mathcal{I}_n$ is a classical DL interpretation with domain $\Delta^{\mathfrak{I}}$: we have $A^{\mathcal{I}_n} \subseteq \Delta^{\mathfrak{I}}$ and $r^{\mathcal{I}_n} \subseteq \Delta^{\mathfrak{I}} \times \Delta^{\mathfrak{I}}$. Rigid roles $r \in \mathsf{N_R^{rig}}$ do not change their interpretation in time: $r^{\mathcal{I}_n} = r^{\mathcal{I}_0}$ for all $n \in \mathbb{Z}$. We usually write $A^{\mathfrak{I},n}$ and $r^{\mathfrak{I},n}$ instead of $A^{\mathcal{I}_n}$ and $r^{\mathcal{I}_n}$, respectively, and the mapping $\cdot^{\mathfrak{I},n}$ is extended to complex $\mathcal{TEL}$-concepts as follows:

$$(C \sqcap D)^{\mathfrak{I},n} = C^{\mathfrak{I},n} \cap D^{\mathfrak{I},n},$$
$$(\exists r.C)^{\mathfrak{I},n} = \{d \mid \text{ there is } e \in C^{\mathfrak{I},n} \text{ with } (d, e) \in r^{\mathfrak{I},n}\},$$
$$(\bigcirc_* C)^{\mathfrak{I},n} = C^{\mathfrak{I}, n \, \mathsf{op}_* \, 1},$$
$$(\Diamond_* C)^{\mathfrak{I},n} = \{d \mid d \in C^{\mathfrak{I}, n \, \mathsf{op}_* \, k} \text{ for some } k > 0\},$$

where op_* stands for $-$ if $* = P$ and for $+$ if $* = F$. We use strict $\Diamond_*$ ($k > 0$) but our results do not depend on the choice.

An interpretation $\mathfrak{I}$ is said to be a *model* of $C \sqsubseteq D$, written $\mathfrak{I} \models C \sqsubseteq D$, if $C^{\mathfrak{I},n} \subseteq D^{\mathfrak{I},n}$, for all $n \in \mathbb{Z}$; and a model of $C \equiv D$ if $C^{\mathfrak{I},n} = D^{\mathfrak{I},n}$, for all $n \in \mathbb{Z}$. We call $\mathfrak{I}$ a *model of a TBox* $\mathcal{T}$, written $\mathfrak{I} \models \mathcal{T}$, if $\mathfrak{I} \models \alpha$ for all $\alpha \in \mathcal{T}$. Note that TBoxes are interpreted *globally* in the sense that all CIs and CDs must be satisfied at every time point. A concept D *subsumes* a concept C with respect to $\mathcal{T}$, written $\mathcal{T} \models C \sqsubseteq D$, if $\mathfrak{I} \models C \sqsubseteq D$ for all models $\mathfrak{I}$ of $\mathcal{T}$.

For ABoxes $\mathcal{A}$ we adopt the *standard name assumption*: $a^{\mathfrak{I},n} = a$ for all $a \in \mathsf{ind}(\mathcal{A})$, $n \in \mathbb{Z}$ (and thus $\mathsf{ind}(\mathcal{A}) \subseteq \Delta^{\mathfrak{I}}$). The relation $\models$ is extended to ABoxes by taking $\mathfrak{I} \models A(a, n)$ iff $a \in A^{\mathfrak{I},n}$ and $\mathfrak{I} \models r(a, b, n)$ iff $(a, b) \in r^{\mathfrak{I},n}$; $\mathfrak{I}$ is a model of $\mathcal{A}$ iff $\mathfrak{I} \models \alpha$ for all $\alpha \in \mathcal{A}$. An interpretation $\mathfrak{I}$ is a *model* of a KB $(\mathcal{T}, \mathcal{A})$, written $\mathfrak{I} \models (\mathcal{T}, \mathcal{A})$, iff $\mathfrak{I} \models \mathcal{T}$ and $\mathfrak{I} \models \mathcal{A}$. Finally, $\mathcal{K} \models A(a, n)$ if $\mathfrak{I} \models A(a, n)$ for every model $\mathfrak{I}$ of $\mathcal{K}$.

As the query language, we consider *temporal atomic queries* (*TAQs*) of the form $A(x, t)$ with $A \in \mathsf{N_C}$, x an *individual variable* and t a *temporal variable*. Given $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, a *certain answer to* $A(x, t)$ *over* $\mathcal{K}$ is a pair $(a, n) \in \mathsf{ind}(\mathcal{A}) \times \mathsf{tem}(\mathcal{A})$ with $\mathcal{K} \models A(a, n)$. We study the complexity of the query answering problem over temporal knowledge bases:

TAQ answering	
Input:	TBox $\mathcal{T}$, ABox $\mathcal{A}$, TAQ $A(x, t)$, a pair (a, n).
Question:	Is (a, n) a certain answer to $A(x, t)$ over $(\mathcal{T}, \mathcal{A})$?

Our results concern both the combined and data complexity of the problem: for data complexity, the TBox is fixed. As usual, for a complexity class $\mathcal{C}$ and a class $\mathcal{X}$ of TBoxes, we say that *TAQ answering over* $\mathcal{X}$ *is* $\mathcal{C}$*-hard in data complexity* if there is some $\mathcal{T} \in \mathcal{X}$ such that answering TAQs over $\mathcal{T}$ is $\mathcal{C}$-hard. Conversely, *TAQ answering over* $\mathcal{X}$ *is in* $\mathcal{C}$ *in data complexity* if, for all $\mathcal{T} \in \mathcal{X}$, answering TAQs over $\mathcal{T}$ is in $\mathcal{C}$.

As classes $\mathcal{X}$, we will in particular look at full $\mathcal{TEL}$ and its fragments $\mathcal{TEL}^{\Diamond}$ and $\mathcal{TEL}^{\bigcirc}$ in which, respectively, only the temporal operators $\Diamond_*$ and $\bigcirc_*$ are allowed. Note that $\Diamond_*$ on the left-hand side and $\Box_*$ (with the usual semantics) on the right-hand side of CIs can be expressed in $\mathcal{TEL}^{\bigcirc}$, e.g., instead of $\Diamond_P A \sqsubseteq X$ or, equivalently, $A \sqsubseteq \Box_F X$, take $A \sqsubseteq A'$ and $\bigcirc_P A' \sqsubseteq A' \sqcap X$, for a fresh A'. Thus, *rigid concepts*, which do not change their interpretation in time, can be expressed in these two fragments using $\Diamond_P \Diamond_F$ on the left-hand side of CIs.

3 Query Answering in $\mathcal{TEL}$: Undecidability

We first pinpoint different sources of complexity for the query answering problem in $\mathcal{TEL}$ in order to identify computationally well-behaved fragments in the sequel.

We begin by showing that TAQ answering over $\mathcal{TEL}^{\diamond}$ is undecidable. The known undecidability of subsumption in $\mathcal{TEL}^{\diamond}$ [Artale et al., 2007] translates only into the combined complexity of TAQ answering. We strengthen the result to obtain undecidability in data complexity by reducing the halting problem for the universal Turing machine. We exploit the crucial observation that disjunction, although not in the syntax, can be simulated with $\diamond_*$ [Artale et al., 2007].

Theorem 1 *TAQ answering over $\mathcal{TEL}^{\diamond}$ is undecidable in data complexity.*

The proof can also be adapted to the *non-strict* semantics of $\diamond_*$ using the *chessboard technique* [Gabbay et al., 2003].

$\mathcal{TEL}^{\circ}$, unlike $\mathcal{TEL}^{\diamond}$, is not capable of expressing disjunction. Still, its expressiveness makes answering TAQs hard:

Theorem 2 *TAQ answering over $\mathcal{TEL}^{\circ}$ is non-elementary in combined complexity and* PSPACE*-hard in data complexity.*

The proof of PSPACE-hardness is close in spirit to that for DATALOG_{1S} [Chomicki and Imielinski, 1988]; note that the lower bound holds even in the restriction of $\mathcal{TEL}^{\circ}$ without $\exists r.C$ on the right-hand side of CIs. For the non-elementary lower bound, we take inspiration in the construction for the product modal logic LTL$\times$**K** [Gabbay et al., 2003, Theorem 6.34]. Our proof requires a careful implementation of the yardstick technique [Stockmeyer, 1974] with only Horn formulas.

Decidability of TAQ answering in full $\mathcal{TEL}^{\circ}$ is left open as interesting and challenging future work; more insights on the difficulty of the problem are given in Sec. 4. Nevertheless, we show that extending $\mathcal{TEL}^{\circ}$ with certain DL constructs that are harmless for data complexity of atemporal query answering [Krisnadhi and Lutz, 2007] immediately leads to undecidability. Let $\mathcal{TELI}^{\circ}$ and $\mathcal{TELF}^{\circ}$ be the extensions of $\mathcal{TEL}^{\circ}$ with *inverse roles* r^- and *functionality* axioms *func*(r), respectively.[1] For both languages, we reduce the halting problem for the universal Turing machine to prove:

Theorem 3 *TAQ answering over $\mathcal{TELI}^{\circ}$ and $\mathcal{TELF}^{\circ}$ is undecidable in data complexity.*

In the rest of the paper, we study decidability and complexity of TAQ answering in various fragments of $\mathcal{TEL}^{\circ}$ and $\mathcal{TEL}^{\diamond}$.

4 Foundations of Query Answering in $\mathcal{TEL}^{\circ}$

In this section, we lay the groundwork for the development of algorithms for query answering in fragments of $\mathcal{TEL}^{\circ}$ by introducing canonical quasimodels, which are succinct abstract representations of the *universal model* of the KB, see also [Artale et al., 2013b; 2015]. They can also be viewed as a generalization of the canonical structures used for query answering in pure $\mathcal{EL}$ [Lutz, Toman, and Wolter, 2009].

In the sequel, we assume that $\mathcal{TEL}^{\circ}$-TBoxes are in *normal form*, that is, they consist of CIs of the form

$$A \sqcap A' \sqsubseteq B, \qquad A \sqsubseteq \exists r.B, \qquad X \sqsubseteq A,$$

where A, A' and B are concept names and X is a *basic concept* of the form A, $\bigcirc_* A$, or $\exists r.A$, for a concept name A. Observe that, without loss of generality, $\bigcirc_*$ is restricted to the left-hand side of CIs: for instance, $A \sqsubseteq \bigcirc_F B$ is equivalent to $\bigcirc_P A \sqsubseteq B$. It is routine to show that every $\mathcal{TEL}^{\circ}$-TBox can be transformed into the normal form by introducing fresh concept names; see, e.g., [Baader, Brandt, and Lutz, 2005].

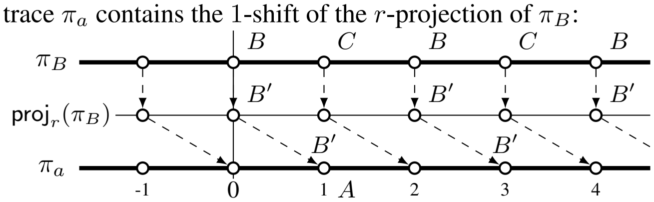
Fix now a KB $(\mathcal{T}, \mathcal{A})$ with a $\mathcal{TEL}^{\circ}$-TBox $\mathcal{T}$ in normal form and let CN be the set of concept names in $(\mathcal{T}, \mathcal{A})$. A map $\pi\colon \mathbb{Z} \to 2^{\mathsf{CN}}$ is a *trace for* $\mathcal{T}$ if it satisfies the following:

(t1) if $A \sqcap A' \sqsubseteq B \in \mathcal{T}$ and $A, A' \in \pi(n)$, then $B \in \pi(n)$;

(t2) if $\bigcirc_* A \sqsubseteq B \in \mathcal{T}$ and $A \in \pi(n)$, then $B \in \pi(n \;\mathsf{op}_*\; 1)$.

Traces are the building blocks of quasimodels: they represent the temporal evolution of individual domain elements. For example, for $\mathcal{T} = \{\bigcirc_P C \sqsubseteq B, \bigcirc_P B \sqsubseteq C\}$, the map π such that $\pi(i)$ is $\{B\}$ for odd i and $\{C\}$ for even i is a trace for $\mathcal{T}$.

In order to describe interactions of domain elements, we require more notation. Let π be a trace for $\mathcal{T}$. For a rigid role $r \in \mathsf{N}_\mathsf{R}^{\mathsf{rig}}$, the *$r$-projection of* π is a map $\mathsf{proj}_r(\pi)\colon \mathbb{Z} \to 2^{\mathsf{CN}}$ that sends each $i \in \mathbb{Z}$ to $\{A \mid \exists r.B \sqsubseteq A \in \mathcal{T}, B \in \pi(i)\}$; for a local role $r \in \mathsf{N}_\mathsf{R}^{\mathsf{loc}}$, $\mathsf{proj}_r(\pi)$ is defined in the same way on 0 but is $\emptyset$ for all other $i \in \mathbb{Z}$. Given a map $\varrho\colon \mathbb{Z} \to 2^{\mathsf{CN}}$ and $n \in \mathbb{Z}$, we say that π *contains the n-shift of* ϱ and write $\varrho \subseteq^n \pi$ if $\varrho(i - n) \subseteq \pi(i)$, for all $i \in \mathbb{Z}$. For example, let $\mathcal{T} = \{\exists r.B \sqsubseteq B'\}$ with rigid role r. In the picture below, the trace π_a contains the 1-shift of the r-projection of π_B:



If r is local then π_a has to contain B' only at 1 (but not at 3, etc.). We are now fully equipped to define quasimodels.

Let $D = \mathsf{ind}(\mathcal{A}) \cup \mathsf{CN}$ henceforth. A *quasimodel $\mathfrak{Q}$ for* $(\mathcal{T}, \mathcal{A})$ is a set of traces π_d, $d \in D$, for $\mathcal{T}$ such that

(q1) $A \in \pi_a(n)$, for all $A(a, n) \in \mathcal{A}$;

(q2) $B \in \pi_B(0)$, for all $B \in \mathsf{CN}$;

(q3) $\mathsf{proj}_r(\pi_b) \subseteq^0 \pi_a$, for all $r(a, b, n) \in \mathcal{A}$;

(q4) if $A \in \pi_d(n)$ then $\mathsf{proj}_r(\pi_B) \subseteq^n \pi_d$, for all $d \in D$, $n \in \mathbb{Z}$ and $A \sqsubseteq \exists r.B$ in $\mathcal{T}$.

Intuitively, quasimodels represent models of $(\mathcal{T}, \mathcal{A})$: each π_a stands for the ABox individual a; each π_B, on the other hand, represents *all* individuals that witness B for CIs $A \sqsubseteq \exists r.B$ in $\mathcal{T}$. The latter is, in fact, the crucial abstraction underlying quasimodels. Note that traces π_B are normalized: B occurs at time point 0, which is compensated by the shift operation in **(q4)**. For example, in the picture above, if $A \sqsubseteq \exists r.B \in \mathcal{T}$ then, in any model, a has an r-successor that belongs to B at moment 1. Such a successor can be obtained as a 'copy' of trace π_B shifted by 1 so that its origin, 0, matches moment 1 for a. Then, by **(q4)**, a belongs to B' at all odd moments.

For the purposes of query answering we need to identify *canonical (minimal)* quasimodels. We define the canonical quasimodel as the limit of the following saturation (chase-like) procedure. Start with initially empty maps π_d, for $d \in D$, and apply **(t1)**–**(t2)**, **(q1)**–**(q4)** as rules: **(q3)**, for example, says 'if

[1] with the usual semantics: $(r^-)^{\mathfrak{I},n} = \{(e, d) \mid (d, e) \in r^{\mathfrak{I},n}\}$; $\mathfrak{I} \models$ *func*(r) iff $e_1 = e_2$, for all $(d, e_1), (d, e_2) \in r^{\mathfrak{I},n}$ and $n \in \mathbb{Z}$.

$r(a,b,n) \in \mathcal{A}$ and $A \in \mathsf{proj}_r(\pi_b)(i)$, then add A to $\pi_a(i)$.' Then we have the following characterization:

Theorem 4 *Let $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ be the canonical quasimodel of $(\mathcal{T}, \mathcal{A})$ with $\mathcal{T}$ a $\mathcal{TEL}^{\circ}$-TBox. Then, for any $A \in \mathsf{CN}$,*

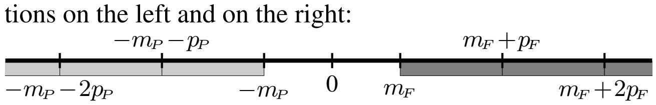
$$(\mathcal{T}, \mathcal{A}) \models A(a,i) \quad \textit{iff} \quad A \in \pi_a(i), \quad \textit{for } a \in \mathsf{ind}(\mathcal{A}),\ i \in \mathbb{Z}.$$

The procedure for constructing the canonical quasimodel deals with infinite data structures (traces) and is generally not terminating. So, although Theorem 4 provides a criterion for certain answers, it does not immediately yield a decision algorithm for full $\mathcal{TEL}^{\circ}$. We remark that known techniques for dealing with such infinite structures cannot be easily applied: for example, MSO (over $\mathbb{Z}$), a standard tool for decidability proofs in temporal DLs [Gabbay et al., 2003], is not sufficient to encode the canonical quasimodel directly because **(q4)** requires $+$. In fact, the key to showing decidability for (fragments of) $\mathcal{TEL}^{\circ}$ is finding a *finite* representation of traces.

The starting point of the rest of the paper is a semantic condition on the canonical quasimodel, *ultimate periodicity*, which ensures decidability in data complexity. Let $\mathcal{T}$ be a $\mathcal{TEL}^{\circ}$-TBox and $\mathfrak{Q}$ the canonical quasimodel for $(\mathcal{T}, \emptyset)$. We say that $\mathcal{T}$ is *ultimately periodic*, if there is $p \in \mathbb{N}$ such that all π_B, $B \in \mathsf{CN}$, in $\mathfrak{Q}$ are *ultimately p-periodic*, that is, for each $B \in \mathsf{CN}$, there are positive integers $m_P, p_P, m_F, p_F \le p$ satisfying the following conditions:

$$\pi_B(n - p_P) = \pi_B(n), \text{ for all } n \le -m_P,$$
$$\pi_B(n + p_F) = \pi_B(n), \text{ for all } n \ge m_F.$$

Intuitively, an ultimately p-periodic trace has repeating sections on the left and on the right:



The condition of ultimate periodicity is rather natural. On the practical side, it is motivated by applications with recurrent patterns such as health care support [Shankar et al., 2008], see CIs (1) and (2) in Section 1. From the theoretical point of view, any satisfiable LTL formula has an ultimately periodic model [Manna and Wolper, 1984].

We next show that ultimate periodicity is indeed sufficient for decidability in data complexity.

Theorem 5 *TAQ answering over ultimately periodic $\mathcal{TEL}^{\circ}$-TBoxes is* PSPACE*-complete in data complexity.*

PSPACE-hardness follows from (the proof of) Theorem 2. We prove the matching upper bound by *rewriting* an ultimately periodic $\mathcal{TEL}^{\circ}$-TBox $\mathcal{T}$ into DATALOG$_{1S}$ [Chomicki and Imielinski, 1988]. First, we take temporal rules

$$r(x,y,t \pm 1) \leftarrow r(x,y,t), \qquad \text{for } r \in \mathsf{N}_\mathsf{R}^{\mathsf{rig}} \text{ in } \mathcal{T}, \tag{3}$$
$$B(x,t) \leftarrow A(x,t), A'(x,t), \qquad \text{for } A \sqcap A' \sqsubseteq B \text{ in } \mathcal{T}, \tag{4}$$
$$B(x,t) \leftarrow r(x,y,t), A(y,t), \qquad \text{for } \exists r.A \sqsubseteq B \text{ in } \mathcal{T}, \tag{5}$$

which reflect rigidity of roles and standard $\mathcal{EL}$ concept inclusions on ABox individuals. Second, we observe that, for any trace π_d in the canonical quasimodel $\mathfrak{Q}$ of any $(\mathcal{T}, \mathcal{A})$, if $A \in \pi_d(n)$ and $A \sqsubseteq \exists r.B \in \mathcal{T}$ then, by **(q4)**, π_d contains the n-shift not only of $\mathsf{proj}_r(\pi_B)$ but also of π_A. Since $\mathcal{T}$ is ultimately periodic, for each trace π_B, we fix integers m_P, p_P, m_F, p_F and take the following rules with a fresh predicate F_B:

$$A(x,t+i) \leftarrow B(x,t), \quad \text{for } 0 \le i < m_F, A \in \pi_B(i),$$
$$A(x,t+i) \leftarrow F_B(x,t), \text{ for } 0 \le i < p_F, A \in \pi_B(m_F + i),$$
$$F_B(x, t+m_F) \leftarrow B(x,t), \qquad F_B(x,t+p_F) \leftarrow F_B(x,t),$$

and symmetric rules with m_P, p_P and fresh P_B. Intuitively, the rules in the first line replicate the (irregular) part of π_B from 0 to m_F. The two rules in the last line add recurring markers F_B at the start of each period while the rules in the second line replicate the period of π_B starting from each marker F_B.

The required DATALOG$_{1S}$-program $\Pi_\mathcal{T}$ contains all the rules above (note that CIs of the form $\bigcirc_* A \sqsubseteq B$ are also covered by the rules for traces π_B). Using the canonical quasimodel and Theorem 4, it is readily seen that $\Pi_\mathcal{T}$ is equivalent to $\mathcal{T}$: for every temporal ABox $\mathcal{A}$, the answers to $\Pi_\mathcal{T}$ over $\mathcal{A}$ coincide with the certain answers to $(\mathcal{T}, \mathcal{A})$. Theorem 5 follows from PSPACE data complexity in DATALOG$_{1S}$ [Chomicki and Imielinski, 1988] and independence of $\Pi_\mathcal{T}$ from $\mathcal{A}$.

Observe that Theorem 5 does not imply decidability of full $\mathcal{TEL}^{\circ}$ since it is open whether every $\mathcal{TEL}^{\circ}$-TBox is ultimately periodic. We thus turn our attention to *sufficient syntactic conditions* for ultimate periodicity and obtain tight complexity bounds for both data and combined complexity for the resulting fragments. We consider two types of conditions: restricted use of rigid roles and acyclicity of concept inclusions.

5 Restricted Use of Rigid Roles

We first consider $\mathcal{TEL}^{\circ}_{\mathsf{loc}}$, the restriction of $\mathcal{TEL}^{\circ}$ in which *only* local roles are allowed. Due to the reduced interaction between the temporal and DL component, we obtain data tractability.

Theorem 6 *TAQ answering over $\mathcal{TEL}^{\circ}_{\mathsf{loc}}$ is* PSPACE*-complete in combined and* PTIME*-complete in data complexity.*

PSPACE-hardness follows from the proof of PSPACE-hardness for entailment in Horn-LTL [Chen and Lin, 1993] and PTIME-hardness from atomic query answering in $\mathcal{EL}$. For the upper bounds, let $(\mathcal{T}, \mathcal{A})$ be a KB with $\mathcal{T}$ a $\mathcal{TEL}^{\circ}_{\mathsf{loc}}$-TBox and $\mathfrak{Q}$ its canonical quasimodel. We take a propositional variable $P_{A,d}$ for each $A \in \mathsf{CN}$ and $d \in D$ and construct a Horn-LTL formula $\varphi_{\mathcal{T},\mathcal{A}}$ whose minimal model is isomorphic to $\mathfrak{Q}$: variable $P_{A,d}$ is true in the model at moment n iff $A \in \pi_d(n)$. We take the conjunction of the following formulas, for $d \in D$:

$$\Box(P_{A,d} \land P_{A',d} \to P_{B,d}), \quad \text{for } A \sqcap A' \sqsubseteq B \in \mathcal{T},$$
$$\Box(\bigcirc_* P_{A,d} \to P_{B,d}), \quad \text{for } \bigcirc_* A \sqsubseteq B \in \mathcal{T},$$
$$\bigcirc^n P_{A,a}, \quad \text{for } A(a,n) \in \mathcal{A},$$
$$P_{B,B}, \quad \text{for } B \in \mathsf{CN},$$
$$\bigcirc^n P_{B,b} \to \bigcirc^n P_{A,a}, \quad \text{for } r(a,b,n) \in \mathcal{A}, \exists r.B \sqsubseteq A \in \mathcal{T},$$
$$P_{B',B} \to \Box(P_{A,d} \to P_{A',d}), \quad \text{for } A \sqsubseteq \exists r.B, \exists r.B' \sqsubseteq A' \in \mathcal{T},$$

where $\bigcirc^n$ is $\bigcirc_F^n$ if $n \ge 0$ and $\bigcirc_P^{-n}$ if $n < 0$ and $\Box$ is the 'globally' operator. It is readily verified that $\varphi_{\mathcal{T},\mathcal{A}}$ is as required. Crucially, **(q4)** for local roles boils down to the last formula above. Since entailment in LTL is in PSPACE [Sistla and Clarke, 1985] and $\varphi_{\mathcal{T},\mathcal{A}}$ is polynomial in the size of $(\mathcal{T}, \mathcal{A})$, we obtain membership in PSPACE for combined complexity.

To obtain the PTime data complexity, observe that traces π_B, $B \in \mathsf{CN}$, are ultimately $2^{|\mathcal{T}|}$-periodic because they are traces of the canonical quasimodel for $(\mathcal{T}, \emptyset)$; so, they can be maintained in constant space. Next, traces π_a, $a \in \mathsf{ind}(\mathcal{A})$, are ultimately $2^{|\mathcal{T}|+|\mathcal{A}|}$-periodic, but a closer inspection reveals that the middle irregular section, $m_P + m_F$, is bounded by $|\mathcal{A}| + 2^{|\mathcal{T}|}$, while both periods, p_P and p_F, by $2^{|\mathcal{T}|}$; see, e.g., Lemma 3 [Artale et al., 2013a]. Thus, $\mathfrak{Q}$ can be stored in space bounded by a polynomial in $|\mathcal{A}|$. Since each rule application extends the traces, the saturation procedure for constructing $\mathfrak{Q}$ terminates in polynomial time in the size of $\mathcal{A}$.

Since ontologies without rigid roles at all may be too restrictive for applications, we consider $\mathcal{TEL}^{\bigcirc}_{\mathsf{l\text{-}rig}}$-TBoxes where rigid roles are allowed only in CIs of the form $\exists r.B \sqsubseteq A$.

Theorem 7 *TAQ answering over $\mathcal{TEL}^{\bigcirc}_{\mathsf{l\text{-}rig}}$ is* PSpace*-complete in data complexity and in* ExpTime *in combined complexity.*

PSpace-hardness in data complexity follows from the proof of Theorem 2. For the upper bounds, we construct rewritings into Datalog_{1S}, similarly to $\Pi_{\mathcal{T}}$ in Section 4 (Theorem 5).

6 Acyclicity Conditions

It is known that acyclicity conditions may lead to better complexity. In particular, acyclic TBoxes are a way of obtaining CTL-based temporal extensions of $\mathcal{EL}$ that have rigid roles and enjoy PTime subsumption [Gutiérrez-Basulto, Jung, and Schneider, 2015]. In Datalog_{1S}, a restriction on recursion has also been used to attain tractability [Chomicki, 1990]. From the application point of view, large parts of Snomed CT and GO [Gene Ontology Cons., 2000] are indeed acyclic. So, we believe that the fragments we consider below are well-suited for temporal extensions of such ontologies.

Acyclic TBoxes are finite sets of CDs $A \equiv C$, $A \in \mathsf{N_C}$, such that no two CDs have the same left-hand side, and there are no CDs $A_1 \equiv C_1, \dots, A_k \equiv C_k$ in $\mathcal{T}$ such that A_{i+1} occurs in C_i, for all $1 \leq i \leq k$ (where $A_{k+1} := A_1$). We say A is *defined in* $\mathcal{T}$ if $A \equiv C \in \mathcal{T}$ and *primitive* otherwise.

Theorem 8 *TAQ answering over acyclic $\mathcal{TEL}^{\bigcirc}$ is in* LogTime*-uniform* AC^0 *in data complexity and in* PTime *in combined complexity.*

The LogTime-uniform AC^0 upper bound is established by *rewriting into FO(+)*: for a given TAQ $A(x, t)$ and TBox $\mathcal{T}$, we construct a two-sorted first-order formula $\varphi_{\mathcal{T},A}(x, t)$ with functions $+1$ and -1 on temporal terms such that $(\mathcal{T}, \mathcal{A}) \models A(a, i)$ iff $\mathcal{A}$ (viewed as an interpretation) is a model of $\varphi_{\mathcal{T},A}(a, i)$, for all ABoxes $\mathcal{A}$, $a \in \mathsf{ind}(\mathcal{A})$, $i \in \mathbb{Z}$. We adapt the technique developed for atemporal $\mathcal{EL}$ [Bienvenu, Lutz, and Wolter, 2012]:

$$\begin{aligned}
\varphi_{\mathcal{T},A}(x,t) &= \mathsf{S}_A(x,t), && \text{if } A \text{ is primitive},\\
\varphi_{\mathcal{T},A}(x,t) &= \mathsf{S}_A(x,t) \lor \varphi_{\mathcal{T},C}(x,t), && \text{if } A \equiv C \in \mathcal{T},\\
\varphi_{\mathcal{T},B_1 \sqcap B_2}(x,t) &= \varphi_{\mathcal{T},B_1}(x,t) \land \varphi_{\mathcal{T},B_2}(x,t), &&\\
\varphi_{\mathcal{T},\exists r.B}(x,t) &= \exists y \big(\mathsf{R}_r(x,y,t) \land \varphi_{\mathcal{T},B}(y,t)\big), &&\\
\varphi_{\mathcal{T},\bigcirc_* B}(x,t) &= \varphi_{\mathcal{T},B}(x, t \mathbin{\mathsf{op}_*} 1), &&
\end{aligned}$$

where $\mathsf{S}_A(x, t)$ is a disjunction of all $B(x, t)$ for concept names B with $\mathcal{T} \models B \sqsubseteq A$, and $\mathsf{R}_r(x, y, t)$ is $r(x, y, t)$ for $r \in \mathsf{N_R^{loc}}$ and $\exists t'\, r(x, y, t')$ for $r \in \mathsf{N_R^{rig}}$.

Note that $\varphi_{\mathcal{T},A}$ is an $\text{FO}^{\mathbb{Z}}$-rewriting in the terminology of Artale et al. [2013b; 2015] because the temporal terms range over $\mathbb{Z}$. However, the infinite interpretation of $\mathcal{A}$ is empty after at most $|\mathcal{T}|$ steps from the ABox and so, $\varphi_{\mathcal{T},A}$ can be converted into an FO-rewriting whose temporal terms range over $\mathsf{tem}(\mathcal{A})$ only; see [Artale et al., 2015].

We next introduce novel notions of acyclicity that restrict only one dimension, DL or temporal.

DL Acyclicity

First, we introduce DL-acyclic $\mathcal{TEL}^{\bigcirc}$-TBoxes, which are well-suited as temporal extensions of, say, biomedical ontologies that may require recurrent patterns but have an acyclic DL component. A $\mathcal{TEL}^{\bigcirc}$-TBox $\mathcal{T}$ with concept names CN is called *DL-acyclic* if there is a mapping $\ell_{\mathsf{DL}} \colon \mathsf{CN} \to \mathbb{N}$ such that:

(i) $A \sqsubseteq \exists r.B$ or $\exists r.B \sqsubseteq A \in \mathcal{T}$ implies $\ell_{\mathsf{DL}}(A) > \ell_{\mathsf{DL}}(B)$;

(ii) $\bigcirc_* A \sqsubseteq B$ implies $\ell_{\mathsf{DL}}(A) = \ell_{\mathsf{DL}}(B)$;

(iii) $A \sqcap A' \sqsubseteq B \in \mathcal{T}$ implies $\ell_{\mathsf{DL}}(A) = \ell_{\mathsf{DL}}(A') = \ell_{\mathsf{DL}}(B)$.

A DL-acyclic TBox is of *depth* k if k is minimal such that a witnessing mapping ℓ_{DL} satisfies $\ell_{\mathsf{DL}}(B) \leq k$ for all $B \in \mathsf{CN}$.

Theorem 9 *TAQ answering over DL-acyclic $\mathcal{TEL}^{\bigcirc}$-TBoxes of depth $k \geq 1$ is k-*ExpSpace*-complete in combined complexity and* NC^1*-complete in data complexity.*

A closer inspection of the non-elementary lower bound proof in Theorem 2 reveals that the TBox used is DL-acyclic and TAQ answering over TBoxes of depth k is k-ExpSpace-hard. NC^1-hardness in data complexity follows [Artale et al., 2015] by reduction of the word problem of NFAs to TAQ answering, even without the DL dimension.

For the matching upper bounds, fix $(\mathcal{T}, \mathcal{A})$ with $\mathcal{T}$ of depth k. We devise a completion procedure, which is based on special LTL-formulas and implies ultimate periodicity of all traces in the canonical quasimodel of $(\mathcal{T}, \mathcal{A})$; cf. Section 5. Given any $\mathcal{A}$, let the slice $\mathcal{A}_i$ consist of all $A(a, i) \in \mathcal{A}$, all $r(a, b, i) \in \mathcal{A}$ and all $r(a, b, i)$ with $r(a, b, j) \in \mathcal{A}$, for some j, and $r \in \mathsf{N_R^{rig}}$. The algorithm separates consequences of the role structure of $\mathcal{A}$ and local temporal consequences of $\mathcal{T}$. In particular, it exhaustively extends $\mathcal{A}$ by all $A(a, i)$ with either

$$(\mathcal{T}, \mathcal{A}_i) \models A(a, i) \;\text{ or }\; B(a, i \mathbin{\mathsf{op}_*} 1) \in \mathcal{A}, \bigcirc_* B \sqsubseteq A \in \mathcal{T}. \quad (6)$$

It turns out that $\mathcal{A}_i$ in (6) can be replaced by its suitably defined *quotient* $\mathcal{B}_i$. Intuitively, the logic can only distinguish distinct trees of depth k, whose number depends on $|\mathcal{T}|$ only; so, the size of $\mathcal{B}_i$ is independent of $|\mathcal{A}|$. By induction on depth k, we define LTL-formulas $\varphi_{a,i}$ of k-fold-exponential size characterizing all $A \in \mathsf{CN}$ with $(\mathcal{T}, \mathcal{B}_i) \models A(a, i)$: we begin from formulas as in Theorem 6; the induction step takes account of the structure of $\mathcal{B}_i$ and incurs an exponential blowup.

Now, for combined complexity, observe that each of the polynomially many $\varphi_{a,i}$ can be analyzed in k-ExpSpace. For data complexity, observe that checking $(\mathcal{T}, \mathcal{B}_i) \models A(a, i)$ can be done in *constant* time. The second option in (6), however, cannot be implemented directly as the number of steps depends on $|\mathcal{A}|$. Instead, we construct a Büchi automaton that accepts precisely the traces for $\mathcal{T}$ and cast the second option in (6) as the question of whether all traces extending $\mathcal{A}$ have A at position i, which is a regular property and so, is in NC^1.

Temporal Acyclicity

We next relax acyclicity by admitting recursion in the DL dimension (but not in temporal); thus, temporally acyclic TBoxes include general $\mathcal{EL}$-TBoxes. A $\mathcal{TEL}^{\bigcirc}$-TBox $\mathcal{T}$ with concept names CN is *temporally acyclic* if there is $\ell_{\bigcirc}\colon \mathsf{CN} \to \mathbb{N}$ such that:

(i) $\bigcirc_{P} A \sqsubseteq B$ or $\bigcirc_{F} B \sqsubseteq A \in \mathcal{T}$ implies $\ell_{\bigcirc}(B) = \ell_{\bigcirc}(A)+1$;
(ii) $\exists r.B \sqsubseteq A$ or $A \sqsubseteq \exists r.B \in \mathcal{T}$ implies $\ell_{\bigcirc}(A) = \ell_{\bigcirc}(B)$;
(iii) $A \sqcap A' \sqsubseteq B \in \mathcal{T}$ implies $\ell_{\bigcirc}(A) = \ell_{\bigcirc}(A') = \ell_{\bigcirc}(B)$.

Temporally acyclic TBoxes cannot, unlike DL acyclic ones, express rigid concepts. Still, we can partition concept names $\mathsf{N_C}$ into local $\mathsf{N_C^{loc}}$ and rigid $\mathsf{N_C^{rig}}$ and obtain the following:

Theorem 10 *TAQ answering over temporally acyclic* $\mathcal{TEL}^{\bigcirc}$ (*with rigid concepts*) *is* PTIME-*complete in data and combined complexity.*

The lower bounds are from $\mathcal{EL}$. For the upper bounds, we show a *small quasimodel property*: traces of the canonical quasimodel of any $(\mathcal{T}, \mathcal{A})$ with such a TBox $\mathcal{T}$ satisfy

$$\pi_a(j) = \pi_a(j'), \qquad \text{if } j, j' > u + |\mathcal{T}| \quad \text{or} \quad j, j' < l - |\mathcal{T}|,$$
$$\pi_B(j) = \pi_B(j'), \qquad \text{if } j, j' > |\mathcal{T}| \quad \text{or} \quad j, j' < -|\mathcal{T}|,$$

where $u = \max \mathcal{A}$ and $l = \min \mathcal{A}$. Intuitively, the canonical quasimodel has a restricted temporal extension that stretches only $|\mathcal{T}|$ time points beyond $\mathcal{A}$. By the small quasimodel property, the procedure for constructing the canonical quasimodel can be implemented in polynomial time: traces π_d require only polynomial space, and rules **(q1)–(q4)** extend the traces.

Inflationary $\mathcal{TEL}^{\diamond}$

Next, we follow an approach suggested by Artale et al. [2013b] (in the context of temporal *DL-Lite*) and restrict $\mathcal{TEL}^{\diamond}$ by allowing $\diamond_{*}$ only on the *left-hand side* of CIs. This fragment is denoted by $\mathcal{TEL}^{\diamond}_{\mathsf{infl}}$, for *inflationary* $\mathcal{TEL}$ (which is related to inflationary DATALOG$_{1S}$ [Chomicki, 1990]). Note that $\mathcal{TEL}^{\diamond}_{\mathsf{infl}}$ extends general $\mathcal{EL}$-TBoxes. Yet, the complexity is the same:

Theorem 11 *TAQ answering over* $\mathcal{TEL}^{\diamond}_{\mathsf{infl}}$ *is* PTIME-*complete in both data and combined complexity.*

We need to show only the upper bounds. Observe that $\mathcal{TEL}^{\diamond}_{\mathsf{infl}}$ can still be viewed as a fragment of $\mathcal{TEL}^{\bigcirc}$; see Section 2. In fact, one can show an analogue of Theorem 4 with the following replacement of **(t2)**:

(t2′) if $\diamond_{*} A \sqsubseteq B \in \mathcal{T}$ and $A \in \pi_d(n)$, then $B \in \pi_d(n')$ for all $n' > n$ if $* = P$ and for all $n' < n$ if $* = F$.

We establish a special shape of the traces in the canonical model of any $(\mathcal{T}, \mathcal{A})$. Let $\varrho\colon \mathbb{Z} \to 2^{\mathsf{CN}}$ be a map and let $l, u \in \mathbb{Z}$ with $l \leq u$. We say that ϱ is an $[l, u]$-*bow tie* if

– for all $i > u$, we have $\varrho(i+1) \supseteq \varrho(i)$, and if $\varrho(i+1) = \varrho(i)$ then all $\varrho(i')$, for $n' \geq i$, coincide;
– symmetrically, for all $i < l$, we have $\varrho(i-1) \supseteq \varrho(i)$, and if $\varrho(i-1) = \varrho(i)$ then all $\varrho(i')$, for $i' \leq i$, coincide.

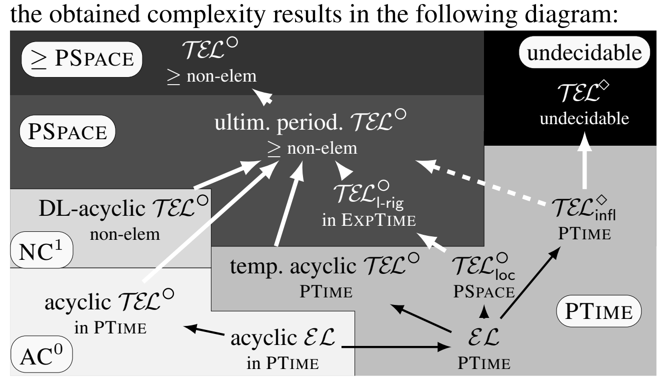
These properties mean that ϱ grows monotonically to the right of u and to the left of l; in other words, ϱ has *inflationary* behaviour. We prove that the traces π_d in the canonical quasimodel $\mathfrak{Q}$ of $(\mathcal{T}, \mathcal{A})$, for any $\mathcal{A}$, enjoy the following properties:

– π_a is a $[\min \mathcal{A}, \max \mathcal{A}]$-bow tie, for each $a \in \mathsf{ind}(\mathcal{A})$;
– π_B is a $[0, 0]$-bow tie, for each $B \in \mathsf{CN}$.

Thus, the traces in $\mathfrak{Q}$ can be represented in polynomial space because only the middle section and at most $|\mathsf{CN}|$ steps at both ends need to be stored. Since the traces are extended with every rule application, the procedure terminates after polynomially many steps; Theorem 11 follows.

7 Discussion and Future Work

We summarize the fragments of $\mathcal{TEL}$, their relationships and the obtained complexity results in the following diagram:



where the solid lines are inclusions between DLs, the dashed line is a reduction that preserves answers to all queries (model conservative extension). The data complexity is indicated by shading and the combined complexity is below the language.

Our data-tractability results show theoretical adequacy of the identified fragments of $\mathcal{TEL}$ for data-intensive applications. Our two novel forms of acyclicity, DL- and temporal, are somewhat close in spirit to *multi-separability* [Chomicki, 1990]: the latter, however, puts a weaker restriction on recursion but a stricter one on the interaction between the temporal and data component. DL-acyclic $\mathcal{TEL}^{\bigcirc}$ is the first (to the best of our knowledge) DL shown to have NC^1-complete query answering (the large gap between data and combined complexity is also remarkable). On the practical side, there is evidence that such data-tractable fragments should be sufficient for many biomedical applications. Following the principles of OBDA, our framework provides a means of defining temporal concepts in the ontology for these applications: temporal concepts capture both (restricted) tree-shaped temporal conjunctive queries (CQs) and recurring temporal patterns.

As our immediate future work, we will address decidability of (full) $\mathcal{TEL}^{\bigcirc}$ and then consider CQs with the $+$ operation on temporal terms. We expect that our positive results can be lifted to CQs using the *combined approach* [Lutz, Toman, and Wolter, 2009], which utilizes a structure similar to our canonical quasimodel. We will also study succinct and expressive representations of temporal data. For example, the only known algorithm for DATALOG$_{1S}$ with *binary* encoding of timestamps in the data runs in EXPTIME in the size of the data [Chomicki and Imielinski, 1988]. We, however, presume that careful materialization should be sufficient to deal with the issue. We will also consider *interval* encoding of temporal ABoxes, e.g., $A(a, [n_1, n_2])$, and settings capturing *infinite* temporal periodic data as introduced by Kabanza, Stévenne, and Wolper [1990] and Chomicki and Imielinski [1993].

References

[Artale et al., 2007] Artale, A.; Kontchakov, R.; Lutz, C.; Wolter, F.; and Zakharyaschev, M. 2007. Temporalising tractable description logics. In *Proc. TIME-07*, 11–22.

[Artale et al., 2013a] Artale, A.; Kontchakov, R.; Ryzhikov, V.; and Zakharyaschev, M. 2013a. The complexity of clausal fragments of LTL. In *Proc. LPAR-19, 2013*, 35–52.

[Artale et al., 2013b] Artale, A.; Kontchakov, R.; Wolter, F.; and Zakharyaschev, M. 2013b. Temporal description logic for ontology-based data access. In *Proc. IJCAI-13*.

[Artale et al., 2015] Artale, A.; Kontchakov, R.; Kovtunova, A.; Ryzhikov, V.; Wolter, F.; and Zakharyaschev, M. 2015. First-order rewritability of temporal ontology-mediated queries. In *Proc. IJCAI-15*, 2706–2712.

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

[Baader, Brandt, and Lutz, 2005] Baader, F.; Brandt, S.; and Lutz, C. 2005. Pushing the $\mathcal{EL}$ envelope. In *Proc. IJCAI-05*.

[Baudinet, Chomicki, and Wolper, 1993] Baudinet, M.; Chomicki, J.; and Wolper, P. 1993. Temporal deductive databases. In *Temporal Databases*. Benjamin/Cummings.

[Bienvenu, Lutz, and Wolter, 2012] Bienvenu, M.; Lutz, C.; and Wolter, F. 2012. Deciding FO-rewritability in $\mathcal{EL}$. In *Proc. DL-12*.

[Borgwardt and Thost, 2015] Borgwardt, S., and Thost, V. 2015. Temporal query answering in the description logic EL. In *Proc. IJCAI-15*, 2819–2825.

[Borgwardt, Lippmann, and Thost, 2015] Borgwardt, S.; Lippmann, M.; and Thost, V. 2015. Temporalizing rewritable query languages over knowledge bases. *J. Web Sem.* 33:50–70.

[Chen and Lin, 1993] Chen, C.-C., and Lin, I.-P. 1993. The computational complexity of satisfiability of temporal Horn formulas in propositional linear-time temporal logic. *Inf. Process. Lett.* 45(3):131–136.

[Chomicki and Imielinski, 1988] Chomicki, J., and Imielinski, T. 1988. Temporal deductive databases and infinite objects. In *Proc. PODS-88*, 61–73.

[Chomicki and Imielinski, 1993] Chomicki, J., and Imielinski, T. 1993. Finite representation of infinite query answers. *ACM Trans. Database Syst.* 18(2):181–223.

[Chomicki and Toman, 2005] Chomicki, J., and Toman, D. 2005. Temporal Databases. In *Handbook of Temporal Reasoning in Artificial Intelligence*, 429–468. Elsevier.

[Chomicki, 1990] Chomicki, J. 1990. Polynomial time query processing in temporal deductive databases. In*Proc. PODS*.

[Dong and Tan, 2015] Dong, X. L., and Tan, W. 2015. A time machine for information: Looking back to look forward. *PVLDB* 8(12):2044–2055.

[Gabbay et al., 2003] Gabbay, D.; Kurucz, A.; Wolter, F.; and Zakharyaschev, M. 2003. *Many-dimensional modal logics: theory and applications*. Elsevier.

[Gene Ontology Cons., 2000] Gene Ontology Cons. 2000. Gene ontology: Tool for the unification of biology. *Nature Genetics* 25:25–29.

[Gutiérrez-Basulto and Klarman, 2012] Gutiérrez-Basulto, V., and Klarman, S. 2012. Towards a unifying approach to representing and querying temporal data in description logics. In *Proc. RR-12*, 90–105.

[Gutiérrez-Basulto, Jung, and Lutz, 2012] Gutiérrez-Basulto, V.; Jung, J. C.; and Lutz, C. 2012. Complexity of branching temporal description logics. In *Proc. ECAI-12*.

[Gutiérrez-Basulto, Jung, and Schneider, 2014] Gutiérrez-Basulto, V.; Jung, J.; and Schneider, T. 2014. Lightweight description logics and branching time: a troublesome marriage. In *Proc. KR-14*.

[Gutiérrez-Basulto, Jung, and Schneider, 2015] Gutiérrez-Basulto, V.; Jung, J.; and Schneider, T. 2015. Lightweight temporal description logics with rigid roles and restricted TBoxes. In *Proc. IJCAI-15*, 3015–3021.

[Haase and Lutz, 2008] Haase, C., and Lutz, C. 2008. Complexity of subsumption in the $\mathcal{EL}$ family of description logics: Acyclic and cyclic TBoxes. In *Proc. of ECAI-08*.

[Kabanza, Stévenne, and Wolper, 1990] Kabanza, F.; Stévenne, J.; and Wolper, P. 1990. Handling infinite temporal data. In *Proc. PODS-90*, 392–403.

[Klarman and Meyer, 2014] Klarman, S., and Meyer, T. 2014. Querying temporal databases via OWL 2 QL. In *Proc. RR-14*, 92–107.

[Krisnadhi and Lutz, 2007] Krisnadhi, A., and Lutz, C. 2007. Data complexity in the $\mathcal{EL}$ family of description logics. In *Proc. LPAR-07*, 333–347.

[Lutz, Toman, and Wolter, 2009] Lutz, C.; Toman, D.; and Wolter, F. 2009. Conjunctive query answering in the description logic $\mathcal{EL}$ using a relational database system. In *Proc. IJCAI-09*.

[Manna and Wolper, 1984] Manna, Z., and Wolper, P. 1984. Synthesis of communicating processes from temporal logic specifications. *ACM Trans.Program.Lang.Syst.* 6(1):68–93.

[O'Connor et al., 2009] O'Connor, M. J.; Shankar, R. D.; Parrish, D. B.; and Das, A. K. 2009. Knowledge-data integration for temporal reasoning in a clinical trial system. *Int. J. Med. Inform.* 78:S77–S85.

[Roth and Tan, 2013] Roth, M., and Tan, W. 2013. Data integration and data exchange: It's really about time. In *Proc. CIDR-13*.

[Shankar et al., 2008] Shankar, R. D.; Martins, S. B.; O'Connor, M. J.; Parrish, D. B.; and Das, A. K. 2008. Representing and reasoning with temporal constraints in clinical trials using semantic technologies. In *BIOSTEC-08, Revised Selected Papers*, 520–530.

[Sistla and Clarke, 1985] Sistla, A. P., and Clarke, E. M. 1985. The complexity of propositional linear temporal logics. *J. ACM* 32(3):733–749.

[Stockmeyer, 1974] Stockmeyer, L. J. 1974. *The Complexity of Decision Problems in Automata Theory and Logic*. Ph.D. Dissertation, MIT, Cambridge, Massachusetts, USA.

A Additional Preliminaries

DATALOG$_{1S}$. We introduce DATALOG$_{1S}$ along the lines of [Chomicki, 1990; Chomicki and Imielinski, 1988]. *Temporal terms* are expressions of the form i, $t \pm i$ where t is a temporal variable and i is a non-negative integer encoded in unary. A *non-temporal term* is a constant or variable name. A *temporal rule* is a formula of the form

$$B(\vec{x}, t) \leftarrow A_1(\vec{x}_1, t_1), \dots, A_k(\vec{x}_k, t_k),$$

where $B, A_1, \dots, A_k$ are predicate symbols, $\vec{x}_i$ are (tuples of) non-temporal terms and $t, t_1, \dots, t_k$ are temporal terms. A DATALOG$_{1S}$ *program* is a finite set of temporal rules. A *temporal database* is a finite set of ground atoms $B(\vec{a}, i)$. Note that temporal databases restricted to predicates of arity two or three are, syntactically, nothing else than temporal ABoxes; thus, we will make no difference between them unless it could cause confusion. Given a set of temporal rules Π, a temporal database D, and a query $A(x, t)$, we say that a pair (a, i) of a constant name a and a time point i is an *answer* and write $(\Pi, \mathcal{A}) \models A(a, i)$ if $A(a, i)$ is true in the *least Herbrand model* $\Pi(\mathcal{A})$ of Π and D, see [Chomicki, 1990]. (The least Herbrand model $\Pi(\mathcal{A})$ can alternatively be defined as the result of exhaustive application of rules in Π to $\mathcal{A}$). It is known that query answering in DATALOG$_{1S}$ is PSPACE-complete in data complexity and EXPTIME-complete in combined complexity [Chomicki and Imielinski, 1988].

Horn-LTL. We introduce syntax and semantics of (a sufficiently expressive version of) propositional Horn-LTL, see also [Chen and Lin, 1993]. Let P be a countably infinite set of propositional variables. Propositional Horn-LTL formulas φ are built according to the following grammar:

$$\begin{aligned} \psi, \psi' &::= p \mid \bigcirc_* \psi \mid \psi \wedge \psi', \\ \varphi, \varphi' &::= \psi \mid \psi \rightarrow \psi' \mid \varphi \wedge \varphi' \mid \Box \varphi', \end{aligned}$$

where $p \in \mathsf{P}$. We often write $\bigcirc^n$ for $\bigcirc_F^n$ if $n \geq 0$ and $\bigcirc_P^{-n}$ for $n < 0$.

A temporal interpretation for a Horn-LTL formulas is a structure $\mathfrak{M} = (W_p)_{p \in \mathsf{P}}$, where $W_p \subseteq \mathbb{Z}$. The relation $\models$ is defined by taking for all $i \in \mathbb{Z}$:

$(\mathfrak{M}, i) \models p,$	if $i \in W_p$;
$(\mathfrak{M}, i) \models \bigcirc_* \psi$	if $(\mathfrak{M}, i \ \mathsf{op}_* \ 1) \models \psi$;
$(\mathfrak{M}, i) \models \varphi_1 \wedge \varphi_2,$	if $(\mathfrak{M}, i) \models \varphi_1$ and $(\mathfrak{M}, i) \models \varphi_2$;
$(\mathfrak{M}, i) \models \varphi_1 \rightarrow \varphi_2,$	if $(\mathfrak{M}, i) \models \varphi_1$ implies $(\mathfrak{M}, i) \models \varphi_2$;
$(\mathfrak{M}, i) \models \Box \varphi,$	if $(\mathfrak{M}, j) \models \varphi$ for all $j \in \mathbb{Z}$.

We say that $\mathfrak{M}$ is a *model* for φ if $(\mathfrak{M}, 0) \models \varphi$. If φ is a propositional Horn-LTL formula and $p \in \mathsf{P}$ then we write $\varphi \models \bigcirc^n p$ to say that $(\mathfrak{M}, n) \models p$, for all $\mathfrak{M}$ with $(\mathfrak{M}, 0) \models \varphi$.

It is folklore that Horn-LTL formulas enjoy a uniquely defined *minimal model*, which, intuitively, makes true only the necessary propositions, that is, the W_p are minimal. It follows that $\varphi \models \bigcirc^n p$ iff $\mathfrak{M}, n \models p$ in the minimal model $\mathfrak{M}$ of φ.

B Proofs

3 Query Answering in $\mathcal{TEL}$: Undecidability

Theorem 1 *TAQ answering over $\mathcal{TEL}^{\Diamond}$ is undecidable in data complexity.*

Proof. The proof is based on the observation that disjunction can be re-introduced in $\mathcal{TEL}^{\Diamond}$. For example, to express $A \sqsubseteq B_1 \sqcup B_2$, we take fresh concept names X, Y and use the four concept inclusions $A \sqsubseteq \Diamond_F X \sqcap \Diamond_F Y$, $\Diamond_F (X \sqcap Y) \sqsubseteq B_1$, $\Diamond_F (X \sqcap \Diamond_F Y) \sqsubseteq B_2$, and $\Diamond_F (Y \sqcap \Diamond_F X) \sqsubseteq B_2$ [Artale et al., 2007]. Together, they imply that each instance of A is an instance of either B_1 or B_2. This observation enables us to use $\sqcup$ in the proof below.

We reduce the (non-)halting problem of the universal Turing machine $\mathfrak{A} = (Q, q_0, \Sigma, \Gamma, \delta, F)$, which we assume without loss of generality to be deterministic. Moreover, we assume that (a) $\mathfrak{A}$ works on a one-end infinite tape, (b) the left-most cell is labeled with a special symbol $\not{c}$ (which is never overwritten), and (c) the right end and everything beyond is labelled with another special symbol, $\not{b}$. It is important to notice that changes between consecutive configurations are *local*: at most three cells around the active cell (under the head of the Turing machine) can change. It will be convenient to say that the current state is *contained* in the active cell. We introduce the following concept names:

- A_a, for $a \in \Gamma$, to represent the contents of the tape;
- Q_i, for $q_i \in Q$, to represent the current state;
- $\overline{Q}$ for marking the non-active cells (the contents of which must be preserved in a transition);
- D^ℓ and D^r for marking the cells immediately left and right of the active cell, respectively;
- I for ensuring that in the tape is blank beyond the input word in the initial configuration;
- F for marking defects;
- A for creating an infinite chain.

A configuration of $\mathfrak{A}$ is represented along the temporal dimension, i.e., by the temporal evolution of a certain individual over time. The (unique) computation of $\mathfrak{A}$ is encoded in the DL dimension along an infinite r-chain, for a rigid role r. The TBox consists of several groups of concept inclusions for different properties of the intended models. Note that we deliberately use disjunction in CIs, even though it is not allowed in the syntax—it is well-known that disjunction can be expressed using $\Diamond_F$ [Artale et al., 2007]. The TBox $\mathcal{T}_{\mathfrak{A}}$ contains the following concept inclusions:

- $A \sqsubseteq \exists r.A$ to create an infinite chain of r-successors;
- $A \sqsubseteq A_{\not{c}}$ to make sure that the cell at the start of the tape is labelled with $\not{c}$ in every configuration;
- $A \sqsubseteq \bigsqcup_i \Diamond_F Q_i$ to choose some state q_i in every configuration;
- $\Diamond_P A \sqsubseteq \bigsqcup_{a \in \Gamma} A_a$ and $\Diamond_P A \sqsubseteq \overline{Q} \sqcup \bigsqcup_{q_i \in Q} Q_i$ to enforce every cell to contain a symbol and either the current state, Q_i, or the no-head marker, $\overline{Q}$;

- $Q_i \sqsubseteq \Diamond_P D^\ell \sqcap \Diamond_F D^r$ to make sure that every state has the two markers around it;
- concept inclusions to identify defects in the encoding of each configuration (by marking them with concept F):

$$\begin{array}{ll} A_a \sqcap A_{a'} \sqsubseteq F, & \text{for all } a \neq a', \\ Q_i \sqcap Q_j \sqsubseteq F, & \text{for all } i \neq j, \\ \overline{Q} \sqcap Q_i \sqsubseteq F, & \text{for all } i, \\ Q_i \sqcap \Diamond_F \Diamond_F D^r \sqsubseteq F, & \text{for all } i, \\ Q_i \sqcap \Diamond_P \Diamond_P D^\ell \sqsubseteq F, & \text{for all } i \end{array}$$

  (the last two CIs ensure that the D^r and D^ℓ markers can only occur immediately next to a Q_i marker);
- concept inclusions to identify defects of transitions between successive configurations (again by marking them with F):
  - non-active cells do not change:

$$\overline{Q} \sqcap A_a \sqcap \exists r.A_{a'} \sqsubseteq F, \quad \text{for all } a \neq a';$$

  - if $\delta(q_i, a) = (q_k, b, \ell)$ then

$$\begin{array}{ll} Q_i \sqcap A_a \sqcap \exists r.A_c \sqsubseteq F, & \text{for } c \neq b, \\ D^\ell \sqcap \Diamond_F (Q_i \sqcap A_a) \sqcap \exists r.Q_j \sqsubseteq F, & \text{for } j \neq k, \\ D^\ell \sqcap \Diamond_F (Q_i \sqcap A_a) \sqcap \exists r.\overline{Q} \sqsubseteq F; & \end{array}$$

    it follows that the only possible state label in the successor configuration is Q_k and the only successor possible symbol is b;
  - similarly, if $\delta(q_i, a) = (q_k, b, r)$ then

$$\begin{array}{ll} Q_i \sqcap A_a \sqcap \exists r.A_c \sqsubseteq F, & \text{for } c \neq b, \\ D^r \sqcap \Diamond_P (Q_i \sqcap A_a) \sqcap \exists r.Q_j \sqsubseteq F, & \text{for } j \neq k, \\ D^r \sqcap \Diamond_P (Q_i \sqcap A_a) \sqcap \exists r.\overline{Q} \sqsubseteq F; & \end{array}$$

- concept inclusions detecting that a non-blank symbol appears to the right of the input word in the initial configuration:

$$I \sqcap \Diamond_F A_a \sqsubseteq F, \quad \text{for all } a \in \Gamma \setminus \{\not b\};$$

- concept inclusions to propagate the defects:

$$\exists r.F \sqsubseteq F \quad \text{and} \quad \Diamond_F F \sqsubseteq F.$$

An input word $w = a_1 \cdots a_n$ is encoded in the ABox $\mathcal{A}_w$ with the following assertions:

$$A(a, 0), Q_0(a, 1), A_{a_1}(a, 1), \dots, A_{a_n}(a, n), I(a, n).$$

Intuitively, any "defect-free" model of $(\mathcal{T}_\mathfrak{A}, \mathcal{A}_w)$ describes an infinite computation of $\mathfrak{A}$ on input w; more precisely, we can show that

$$(\mathcal{T}_\mathfrak{A}, \mathcal{A}_w) \models F(a, 0) \qquad \text{iff} \qquad \mathfrak{A} \text{ halts on input } w.$$

This completes the proof of Theorem 1. ❏

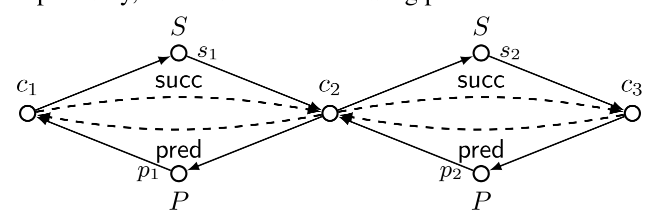
We defer the proof of Theorem 2 until the end of the section and present first proofs that are more similar in the structure to the proof of Theorem 1.

Lemma 12 *There is a $\mathcal{TEL}^\bigcirc$-TBox $\mathcal{T}$ using only a single role name $r \in \mathsf{N}_\mathsf{R}^{\mathsf{rig}}$ and no $\exists r.B$ on the right-hand side of CIs such that TAQ answering over $\mathcal{T}$ is* PSPACE*-hard in data complexity.*

Proof. We reduce the word problem for deterministic linear space-bounded Turing machines (LBAs), similarly to [Chomicki, 1990]. Let $\mathfrak{A} = (Q, \Sigma, \Gamma, \delta, q_0, F)$ be an LBA with a PSPACE-hard word problem. The TBox, $\mathcal{T}_\mathfrak{A}$, uses the following concept names (cf. the proof of Theorem 1):

- A_a, for each $a \in \Gamma$, to encode the content of a tape cell;
- Q_i, for $q_i \in Q$, to represent the current state;
- $\overline{Q}$ for marking the no-head cells; in addition, we use $\overline{Q}^\ell$ and $\overline{Q}^r$ for the non-head cells on the left and right of the active cell, respectively;
- G to propagate an accepting state.

We also use a single rigid role name $r \in \mathsf{N}_\mathsf{R}^{\mathsf{rig}}$. For the sake of readability, we introduce two auxiliary concept names, P and S, and abbreviate the complex concepts $\exists r.(P \sqcap \exists r.C)$ and $\exists r.(S \sqcap \exists r.C)$ by $\exists \mathsf{pred}.C$ and $\exists \mathsf{succ}.C$, respectively. A configuration of $\mathfrak{A}$ is encoded in the DL dimension: neighbouring cells are connected back-and-forth via succ and pred, respectively, as shown in the following picture:



The TBox $\mathcal{T}_\mathfrak{A}$ consists now of the following set of concept inclusions. For a transition $\delta(q_i, a) = (q_k, b, r)$, we take

$$\begin{aligned} \exists \mathsf{pred}.(Q_i \sqcap A_a) &\sqsubseteq \bigcirc_F Q_k, \\ Q_i \sqcap A_a &\sqsubseteq \bigcirc_F A_b. \end{aligned}$$

For a transition $\delta(q_i, a) = (q_k, b, l)$, we take

$$\begin{aligned} \exists \mathsf{succ}.(Q_i \sqcap A_a) &\sqsubseteq \bigcirc_F Q_k, \\ Q_i \sqcap A_a &\sqsubseteq \bigcirc_F A_b. \end{aligned}$$

The content of non-active cells is preserved by concept inclusions, for $a \in \Gamma$,

$$\overline{Q} \sqcap A_a \sqsubseteq \bigcirc_F A_a$$

and the following concept inclusions that propagate the no-head markers along the tape:

$$\begin{array}{lll} \exists \mathsf{pred}.Q_i \sqsubseteq \overline{Q}^r, & \exists \mathsf{succ}.Q_i \sqsubseteq \overline{Q}^\ell, & \text{for } q_i \in Q, \\ \exists \mathsf{pred}.\overline{Q}^r \sqsubseteq \overline{Q}^r, & \exists \mathsf{succ}.\overline{Q}^\ell \sqsubseteq \overline{Q}^\ell, & \\ \overline{Q}^r \sqsubseteq \overline{Q}, & \overline{Q}^\ell \sqsubseteq \overline{Q}. & \end{array}$$

The auxiliary concept names S and P are simply propagated into the future:

$$S \sqsubseteq \bigcirc_F S \qquad P \sqsubseteq \bigcirc_F P.$$

When an accepting state is reached, we activate the marker G and propagate it back to the left-most cell at moment 0:

$$\begin{aligned} Q_i &\sqsubseteq G, \qquad \text{for } q_i \in F,\\ \bigcirc_{F} G &\sqsubseteq G,\\ \exists \mathsf{succ}.G &\sqsubseteq G. \end{aligned}$$

An input word $w = a_1 \cdots a_n$ is encoded as an ABox $\mathcal{A}_w$ with $\mathsf{ind}(\mathcal{A}_w) = \{c_i \mid 1 \leq i \leq n\} \cup \{p_i, s_i \mid 1 \leq i < n\}$ and the following assertions that initialize the tape and fix the succ and pred relations:

$$\begin{array}{ll} Q_0(c_1, 0),\ A_{a_i}(c_i, 0), & \text{for } 1 \leq i \leq n,\\ r(c_i, s_i, 0), S(s_i, 0), r(s_i, c_{i+1}, 0), & \text{for } 1 \leq i < n,\\ r(c_{i+1}, p_i, 0), P(p_i, 0), r(p_i, c_i, 0), & \text{for } 1 \leq i < n. \end{array}$$

It is routine to verify that $\mathfrak{A}$ accepts input w just in case $(\mathcal{T}_{\mathfrak{A}}, \mathcal{A}_w) \models G(c_1, 0)$. ❑

Theorem 3 *TAQ answering over $\mathcal{TELI}^{\bigcirc}$ and $\mathcal{TELF}$ is undecidable in data complexity.*

Proof. We give a proof only for $\mathcal{TELI}^{\bigcirc}$, a proof for $\mathcal{TELF}$ is a straightforward adaptation, which is indicated below.

We reduce the halting problem of the universal Turing machine $\mathfrak{A} = (Q, q_0, \Sigma, \Gamma, \delta, F)$, which, similarly to the proof of Theorem 1, is assumed to be deterministic. Again, we assume that $\mathfrak{A}$ works on a one-end infinite tape and that the left-most cell is labelled with a special marker $\not c$ (which is never overwritten) and the right end and everything beyond is labelled with another special symbol, $\not b$. We use the following concept names, with the intuition similar to the proof of Lemma 12:

– A_a, for $a \in \Gamma$, to represent contents of the tape;

– Q_i, for $q_i \in Q$, to represent the current state;

– $\overline{Q}$ for marking the no-head cells; $\overline{Q}^{\ell}$ and $\overline{Q}^{r}$ are used for the non-head cells on the left and right of the active cell, respectively;

– G for propagating an accepting states.

(Note, however, that unlike in the proof of Lemma 12, we use the temporal dimension to represent configurations and the DL dimension for transitions between the configurations.)

As in the proof of Theorem 1, a configuration of $\mathfrak{A}$ is represented along the temporal dimension, that is, by the temporal evolution of a certain individual over time. The computation of $\mathfrak{A}$ is done in the DL dimension along an infinite r-chain, for a rigid role name r.

We start by enforcing that everything right of $\not b$ is labeled with $\not b$ using concept inclusion

$$A_{\not b} \sqsubseteq \bigcirc_{F} A_{\not b}.$$

We next introduce the constraints needed to synchronise two successive configurations. To this end, we enforce every cell left and right of the state to be labelled by $\overline{Q}$ by adding

$$\begin{array}{lll} \bigcirc_{P} Q_i \sqsubseteq \overline{Q}^{r}, & \bigcirc_{F} Q_i \sqsubseteq \overline{Q}^{\ell}, & \text{for all } q_i \in Q,\\ \bigcirc_{P} \overline{Q}^{r} \sqsubseteq \overline{Q}^{r}, & \bigcirc_{F} \overline{Q}^{\ell} \sqsubseteq \overline{Q}^{\ell}, & \\ \overline{Q}^{r} \sqsubseteq \overline{Q}, & \overline{Q}^{\ell} \sqsubseteq \overline{Q}. & \end{array}$$

For transitions between successive configuration, we include the following concept inclusions:

$$\begin{array}{ll} Q_i \sqcap A_a \sqsubseteq \exists r.(A_b \sqcap \bigcirc_{F} Q_k), & \text{if } \delta(q_i, a) = (q_k, b, r),\\ Q_i \sqcap A_a \sqsubseteq \exists r.(A_b \sqcap \bigcirc_{P} Q_k), & \text{if } \delta(q_i, a) = (q_k, b, \ell). \end{array}$$

Non-active cells (those labelled by $\overline{Q}$) are not changed, so we add, for all $a \in \Gamma$:

$$\exists r^{-}.(\overline{Q} \sqcap A_a) \sqsubseteq A_a.$$

(Note that this is the only concept inclusion that uses inverse roles. If r is functional, we can use $\overline{Q} \sqcap A_a \sqsubseteq \exists r.A_a$ instead.) Finally, for identifying an accepting configuration, we include

$$\begin{aligned} Q_i &\sqsubseteq G, \qquad \text{for } q_i \in F,\\ \exists r.G &\sqsubseteq G,\\ \bigcirc_{F} G &\sqsubseteq G. \end{aligned}$$

The ABox $\mathcal{A}_w$ encodes an input word $w = a_1 \cdots a_n$ as the following assertions:

$$A(a, 0), Q_0(a, 1), A_{a_1}(a, 1), \ldots, A_{a_n}(a, n), \qquad A_{\not b}(a, n+1).$$

It is now routine to verify that $(\mathcal{T}_{\mathfrak{A}}, \mathcal{A}_w) \models G(a, 0)$ iff $\mathfrak{A}$ accepts input w. ❑

Theorem 2 *TAQ answering over $\mathcal{TEL}^{\bigcirc}$ is non-elementary in combined complexity and* PSPACE*-hard in data complexity.*

Proof. PSPACE-hardness is by Lemma 12. For the non-elementary lower bound we take inspiration in the construction for the product modal logic LTL$\times$**K** [Gabbay et al., 2003, Theorem 6.34]. Our proof requires a careful implementation of the yardstick technique [Stockmeyer, 1974] using only Horn formulas. The proof consists of two steps: first, we show how to encode arbitrarily large elementary numbers; second, we prove k-EXPSPACE-hardness for every $k \geq 1$. We concentrate on the first step, because, given the encoding of the numbers, it is straightforward to establish the respective lower bounds. We define function $\exp_k$ by taking

$$\exp_0(n) = n, \qquad \exp_{k+1}(n) = \exp_k(n) \cdot 2^{\exp_k(n)}.$$

Lemma 13 *For each $k, n \geq 1$, there is a DL-acyclic $\mathcal{TEL}^{\bigcirc}$-TBox $\mathcal{T}_{k,n}$, and concept names $\mathsf{Init}_k, \mathsf{Zero}_k, \overline{\mathsf{Zero}_k}$ such that*

$$\mathcal{T}_{k,n} \models \mathsf{Init}_k \sqsubseteq \bigcirc_{F}^{m} \mathsf{Zero}_k \ \textit{ iff } \ m = 0 \bmod \exp_k(n), \quad (7)$$

$$\mathcal{T}_{k,n} \models \mathsf{Init}_k \sqsubseteq \bigcirc_{F}^{m} \overline{\mathsf{Zero}_k} \ \textit{ iff } \ m \neq 0 \bmod \exp_k(n). \quad (8)$$

Proof. Fix $n \geq 1$. We use single *rigid* role r and throughout the proof, concepts of the form $\overline{A}$ will be intended as the complements of respective A.

The proof is by induction on k. For the basis of induction, $k = 1$, we describe a $\mathcal{TEL}^{\bigcirc}$-TBox $\mathcal{T}_{1,n}$ that models a binary counter with n bits along the temporal dimension (these will be counters at level 1). The TBox is based on the following concept names:

– S_1 and $\overline{S}_1$ to indicate the start of the encoding of the counter,

– B_1 and $\overline{B}_1$ to represent the bits of the binary counter;

– C_1 and $\overline{C}_1$ for the value of the carry bit;

– $\mathsf{Init}_1, \mathsf{Zero}_1$ and $\overline{\mathsf{Zero}}_1$ for the claim of the lemma.

First, $\mathcal{T}_{1,n}$ sets out delimiters each n moments of time:

$$\mathsf{Init}_1 \sqsubseteq S_1, \tag{9}$$
$$S_1 \sqsubseteq \bigcirc_F^n S_1, \tag{10}$$
$$S_1 \sqsubseteq \bigcirc_F^i \overline{S}_1, \text{ for } 1 \leq i < n. \tag{11}$$

Then, we write the representation of 0 as the first value of the counter:

$$\mathsf{Init}_1 \sqsubseteq \overline{B}_1 \sqcap \bigcirc_F^1 \overline{B}_1 \sqcap \ldots \sqcap \bigcirc_F^n \overline{B}_1. \tag{12}$$

The following CIs increment the value of the n-bit counter (with the least significant bit at the smallest time point):

$$S_1 \sqcap \bigcirc_P^n B_1 \sqsubseteq \overline{B}_1 \sqcap \bigcirc_F C_1, \tag{13}$$
$$S_1 \sqcap \bigcirc_P^n \overline{B}_1 \sqsubseteq B_1 \sqcap \bigcirc_F \overline{C}_1, \tag{14}$$
$$\overline{S}_1 \sqcap C_1 \sqcap \bigcirc_P^n B_1 \sqsubseteq \overline{B}_1 \sqcap \bigcirc_F C_1, \tag{15}$$
$$\overline{S}_1 \sqcap C_1 \sqcap \bigcirc_P^n \overline{B}_1 \sqsubseteq B_1 \sqcap \bigcirc_F \overline{C}_1, \tag{16}$$
$$\overline{S}_1 \sqcap \overline{C}_1 \sqcap \bigcirc_P^n B_1 \sqsubseteq B_1 \sqcap \bigcirc_F \overline{C}_1, \tag{17}$$
$$\overline{S}_1 \sqcap \overline{C}_1 \sqcap \bigcirc_P^n \overline{B}_1 \sqsubseteq \overline{B}_1 \sqcap \bigcirc_F \overline{C}_1. \tag{18}$$

Intuitively, S_1 marks the positions for the least significant bit in the binary counter. Then, (13)–(16) flip all the bits until 0 is reached. After that, (18) and (17) copy the remaining bits (until the delimiter, S_1, is reached). Concept names Zero_1 and $\overline{\mathsf{Zero}}_1$ are then related to the binary representation of the counter by means of

$$S_1 \sqcap \overline{B}_1 \sqcap \ldots \sqcap \bigcirc_F^n \overline{B}_1 \sqsubseteq \mathsf{Zero}_1, \tag{19}$$
$$S_1 \sqcap \bigcirc_F^i B_1 \sqsubseteq \overline{\mathsf{Zero}}_1, \quad \text{for } 0 \leq i < n. \tag{20}$$

It should be clear that $\mathcal{T}_{1,n}$ satisfies (7) and (8). We only remark that, that due to the structure of the CIs above, if $d \in \mathsf{Init}_1^{\mathcal{J},t}$, for a model $\mathcal{J}$ of $\mathcal{T}_{1,n}$, then at each moment $m \geq t$, the element d belongs either to A or $\overline{A}$, for all concepts A of the form B_1, C_1 and S_1. On the other hand, there is always a model of $\mathcal{T}_{1,n}$ in which, for all concepts names A as above, none of the domain elements belongs to both A and $\overline{A}$.

For the inductive step, let $k > 1$. By the induction hypothesis, there is a TBox $\mathcal{T}_{k-1,n}$ satisfying (7) and (8) for $k-1$. So, $\mathcal{T}_{k-1,n}$ is the first ingredient of $\mathcal{T}_{k,n}$: let $\mathcal{T}_{k,n}$ contain all concept inclusions of $\mathcal{T}_{k-1,n}$. As before, a counter value is represented by a sequence of $\exp_{k-1}(n)$ consecutive time points but the main difficulty is now to relate two consecutive counter values. For that purpose, we use the following symbols (in addition to the signature of $\mathcal{T}_{k-1,n}$):

– S_k and $\overline{S}_k$ to indicate the start of the encoding of the counter value at level k,

– Z_k to write down the initial counter value of 0,

– B_k and $\overline{B}_k$ to represent the bits of the binary counter,

– C_k and $\overline{C}_k$ and for the value of the carry bit,

– Y_k and $\overline{Y}_k$ to record the value of the bit and communicate it between two consecutive numbers,

– I_k and $\overline{I}_k$ to detect sequences of set bits of length $\exp_{k-1}(n)$,

– $\mathsf{Delim}_k, \mathsf{Keep}_k, \mathsf{Read}_k$ to distinguish different uses of the counters.

– $\mathsf{Init}_k, \mathsf{Zero}_k$ and $\overline{\mathsf{Zero}}_k$ as required in claim of the lemma.

Intuitively, the level k counter is realized by first enforcing a sequence of $\exp_{k-1}(n)$ zeros, that is, $\overline{B}_k$, along the temporal dimension, and then enforcing that, if there is a sequence of length $\exp_{k-1}(n)$ encoding some number, say M, then the subsequent sequence of the same length encodes $M+1$. We mark the beginnings of a number using concepts S_k and $\overline{S}_k$ and a special counter of type Delim_k of level $k-1$ as follows:

$$\mathsf{Init}_k \sqsubseteq \exists r.(\mathsf{Delim}_k \sqcap \mathsf{Init}_{k-1}), \tag{21}$$
$$\bigcirc_P \mathsf{Delim}_k \sqsubseteq \mathsf{Delim}_k, \tag{22}$$
$$\exists r.(\mathsf{Delim}_k \sqcap \mathsf{Zero}_{k-1}) \sqsubseteq S_k, \tag{23}$$
$$\exists r.(\mathsf{Delim}_k \sqcap \overline{\mathsf{Zero}}_{k-1}) \sqsubseteq \overline{S}_k; \tag{24}$$

cf. (9)–(11). Next, we write down 0 as the first value of the counter, that is, we enforce the first $\exp_{k-1}(n)$ time points belong to $\overline{B}_k$:

$$\mathsf{Init}_k \sqsubseteq Z_k, \tag{25}$$
$$\overline{S}_k \sqcap \bigcirc_P Z_k \sqsubseteq Z_k, \tag{26}$$
$$Z_k \sqsubseteq \overline{B_k}; \tag{27}$$

cf. (12). In order to correctly increment the level k counter, we have to communicate between time point elements having a distance of $\exp_{k-1}(n)$. The following concept inclusions create a individual and store the value of the current bit there:

$$B_k \sqsubseteq \exists r.(\mathsf{Keep}_k \sqcap \mathsf{Init}_{k-1} \sqcap Y_k), \tag{28}$$
$$\overline{B}_k \sqsubseteq \exists r.(\mathsf{Keep}_k \sqcap \mathsf{Init}_{k-1} \sqcap \overline{Y}_k), \tag{29}$$
$$\bigcirc_P Y_k \sqsubseteq Y_k, \tag{30}$$
$$\bigcirc_P \overline{Y}_k \sqsubseteq \overline{Y}_k. \tag{31}$$

In order to transfer the stored value, Y_k or $\overline{Y}_k$, *precisely* $\exp_{k-1}(n)$ time points, the relevant time point is marked with Read_k:

$$\overline{\mathsf{Zero}}_{k-1} \sqcap \bigcirc_P \mathsf{Keep}_k \sqsubseteq \mathsf{Keep}_k, \tag{32}$$
$$\mathsf{Zero}_{k-1} \sqcap \bigcirc_P \mathsf{Keep}_k \sqsubseteq \mathsf{Read}_k. \tag{33}$$

Intuitively, when an r-successor is created, the first $\exp_{k-1}(n) - 1$ moments (from the current one) are marked by Keep_k and the $\exp_{k-1}(n)$ moment is marked by Read_k. Finally, the next group of CIs realizes the increment of a level k counter:

$$S_k \sqcap \exists r.(\mathsf{Read}_k \sqcap Y_k) \sqsubseteq \overline{B}_k \sqcap \bigcirc_F C_k, \tag{34}$$
$$S_k \sqcap \exists r.(\mathsf{Read}_k \sqcap \overline{Y}_k) \sqsubseteq B_k \sqcap \bigcirc_F \overline{C}_k, \tag{35}$$
$$\overline{S}_k \sqcap C_k \sqcap \exists r.(\mathsf{Read}_k \sqcap Y_k) \sqsubseteq \overline{B}_k \sqcap \bigcirc_F C_k, \tag{36}$$
$$\overline{S}_k \sqcap C_k \sqcap \exists r.(\mathsf{Read}_k \sqcap \overline{Y}_k) \sqsubseteq B_k \sqcap \bigcirc_F \overline{C}_k, \tag{37}$$
$$\overline{S}_k \sqcap \overline{C}_k \sqcap \exists r.(\mathsf{Read}_k \sqcap Y_k) \sqsubseteq B_k \sqcap \bigcirc_F \overline{C}_k, \tag{38}$$

$$\overline{S}_k \sqcap \overline{C}_k \sqcap \exists r.(\mathsf{Read}_k \sqcap \overline{Y}_k) \sqsubseteq \overline{B}_k \sqcap \bigcirc_F \overline{C}_k. \tag{39}$$

These CIs work in precisely the same way as (13)–(18) for the case of $k = 1$ except that now we use $\exists r.(\mathsf{Read}_k \sqcap Y_k)$ and $\exists r.(\mathsf{Read}_k \sqcap \overline{Y}_k)$ in place of $\bigcirc_P^n B_1$ and $\bigcirc_P^n \overline{B}_1$.

To obtain correct valuations for concepts Zero_k and $\overline{\mathsf{Zero}}_k$, we require an auxiliary concept I_k defined by the following set of CIs:

$$S_k \sqcap B_k \sqsubseteq I_k, \tag{40}$$
$$\overline{B}_k \sqsubseteq \overline{I}_k, \tag{41}$$
$$\overline{S}_k \sqcap \bigcirc_P I_k \sqcap B_k \sqsubseteq I_k, \tag{42}$$
$$\overline{S}_k \sqcap \bigcirc_P \overline{I}_k \sqsubseteq \overline{I}_k. \tag{43}$$

Informally, I_k identifies long runs of set bits that begin at one of the delimiters. With I_k and $\overline{I}_k$ at hand, we add the following CIs to define Zero_k and $\overline{\mathsf{Zero}}_k$:

$$\mathsf{Init}_k \sqsubseteq \mathsf{Zero}_k, \tag{44}$$
$$S_k \sqcap \bigcirc_P I_k \sqsubseteq \mathsf{Zero}_k, \tag{45}$$
$$S_k \sqcap \bigcirc_P \overline{I}_k \sqsubseteq \overline{\mathsf{Zero}}_k, \tag{46}$$
$$\overline{S}_k \sqsubseteq \overline{\mathsf{Zero}}_k. \tag{47}$$

This completes the construction of $\mathcal{T}_{k,n}$ and it remains to show (7) and (8).

Let $\mathfrak{I}$ be a model of $\mathcal{T}_{k,n}$ and $d \in \mathsf{Init}_k^{\mathfrak{I},t}$. Since the arguments are independent of t, we will assume $t = 0$ from now on. Let $N = \exp_{k-1}(n)$. By (21), there is some $e \in \Delta^{\mathfrak{I}}$ such that[2] $(d, e) \in r^{\mathfrak{I}}$ and $e \in \mathsf{Init}_{k-1}^{\mathfrak{I},0}, \mathsf{Delim}_k^{\mathfrak{I},0}$. By (22), we have $e \in \mathsf{Delim}_k^{\mathfrak{I},i}$ for all $i \geq 0$; by (7) and (8) of the induction hypothesis, we have, for all $i \geq 0$,

$$e \in \mathsf{Zero}_{k-1}^{\mathfrak{I},i} \quad \text{if} \quad i = 0 \bmod N,$$
$$e \in \overline{\mathsf{Zero}}_{k-1}^{\mathfrak{I},i} \quad \text{if} \quad i \neq 0 \bmod N.$$

By (23) and (24), we obtain

$$d \in S_k^{\mathfrak{I},i} \quad \text{if} \quad i = 0 \bmod N, \tag{48}$$
$$d \in \overline{S}_k^{\mathfrak{I},i} \quad \text{if} \quad i \neq 0 \bmod N. \tag{49}$$

For each $i \geq 0$, define $\mathsf{bit}(i)$ as follows:

$$\mathsf{bit}(i) = \begin{cases} \{0\} & \text{if } d \in \overline{B}_k^{\mathfrak{I},i} \setminus B_k^{\mathfrak{I},i}; \\ \{1\} & \text{if } d \in B_k^{\mathfrak{I},i} \setminus \overline{B}_k^{\mathfrak{I},i}; \\ \{0,1\} & \text{otherwise.} \end{cases}$$

Starting from that, we define for each $m \geq 0$ a function $\mathsf{val}(m)$ reading off the m-th counter value:

$$\mathsf{val}(m) = \Big\{ \sum_{i=0}^{N-1} 2^i b_i \mid b_i \in \mathsf{bit}(mN + i) \Big\}.$$

Clearly, $\mathsf{val}(m) \subseteq \{0, \dots, 2^N - 1\}$.

Claim 1. $m \bmod 2^N \in \mathsf{val}(m)$, for all $m \geq 0$.

[2] Since r is rigid, we will not specify the moment of time.

Proof of Claim 1. The proof is by induction on m. For the induction base, let $m = 0$. Clearly, it suffices to show $0 \in \mathsf{bit}(i)$, for every $0 \leq i < N$. By (25)–(27) and (49), $d \in \overline{B}_k^{\mathfrak{I},i}$ for all $0 \leq i < N$.

For the induction step, take $m > 0$. By induction, we know that $m - 1 \bmod 2^N \in \mathsf{val}(m-1)$. We show the following characterization of a single incrementation operation via an inductive argument for all $mN \leq i < (m+1)N$:

(P1) If $1 \in \mathsf{bit}(j)$ for all $mN \leq j < i$, then $d \in C_k^{\mathfrak{I},i}$; otherwise, $d \in \overline{C}_k^{\mathfrak{I},i}$;

(P2) If $d \in C_k^{\mathfrak{I},i}$, then $\{1 - v \mid v \in \mathsf{bit}(i - N)\} \subseteq \mathsf{bit}(i)$;

(P3) If $d \in \overline{C}_k^{\mathfrak{I},i}$, then $\mathsf{bit}(i - N) \subseteq \mathsf{bit}(i)$.

By (34) and (48), $d \in C_k^{\mathfrak{I},mN}$. If $1 \in \mathsf{bit}((m-1)N)$, then $d \in B_k^{\mathfrak{I},(m-1)N}$. By (28), there is some e with $(d, e) \in r^{\mathfrak{I}}$ and $e \in (\mathsf{Init}_{k-1} \sqcap \mathsf{Keep}_k \sqcap Y_k)^{\mathfrak{I},(m-1)N}$. By (30), $e \in Y_k^{\mathfrak{I},mN}$. Then, (32)–(33) together with (7) and (8) of the induction hypothesis imply $e \in \mathsf{Read}_k^{\mathfrak{I},mN}$, whence $d \in \exists r.(\mathsf{Read}_k \sqcap Y_k)^{\mathfrak{I},mN}$. By (36), $d \in \overline{B}^{\mathfrak{I},mN}$, and hence $0 \in \mathsf{bit}(mN)$. The carry bit is also propagated to $mN + 1$. The transfer of $\overline{Y}_k$ in case $0 \in \mathsf{bit}((m-1)N)$ is analogous, that is, (37) implies $0 \in \mathsf{bit}(mN)$ and $d \in \overline{C}_k^{\mathfrak{I},mN+1}$.

The remaining cases are treated in a similar way. Then, Properties **(P2)** and **(P3)** imply that

$$\big\{ v + 1 \bmod 2^N \mid v \in \mathsf{val}(m-1) \big\} \subseteq \mathsf{val}(m),$$

which finishes the proof of Claim 1.

We next show that the maximal value, $2^N - 1$, is recognized correctly.

Claim 2. For all $m \geq 0$,

– if $2^N - 1 \in \mathsf{val}(m)$, then $d \in I_k^{\mathfrak{I},(m+1)N-1}$;

– if $k \in \mathsf{val}(m)$ for $k \neq 2^N - 1$, then $d \in \overline{I}_k^{\mathfrak{I},(m+1)N-1}$.

Proof of Claim 2. Suppose first that $2^N - 1 \in \mathsf{val}(m)$, that is, for each $0 \leq i < N$, we have $1 \in \mathsf{bit}(mN + i)$ and thus $d \in B_k^{\mathfrak{I},mN+i}$. By (40) and (48), we obtain $d \in I_k^{\mathfrak{I},mN}$. CI (42) with (49) yield $d \in I_k^{\mathfrak{I},mN+i}$ for all $0 \leq i < N$.

Suppose now that $k \in \mathsf{val}(m)$ with $k \neq 2^N - 1$, that is, there is some $0 \leq i < N$ with $0 \in \mathsf{bit}(mN + i)$ and thus $d \in \overline{B}_k^{\mathfrak{I},mN+i}$. By (41), $d \in \overline{I}_k^{\mathfrak{I},mN+i}$. By (43) and (49), $d \in \overline{I}_k^{\mathfrak{I},mN+j}$, for all $i \leq j < N$. This finishes the proof of Claim 2.

We are now in a position to show (7) and (8) for k. We concentrate on the "if"-directions; the "only if"-directions are consequences of the observation that there is a *minimal* model of $\mathcal{T}_{k,n}$ and $\mathsf{Init_k}$ such that, for all concept names A, none of the domain elements belongs to both A and $\overline{A}$ (at any time point).

If $m \neq 0 \bmod N$, then $d \in \overline{S}_k^{\mathfrak{I},m}$ by (49). By (47), $d \in \overline{\mathsf{Zero}}_k^{\mathfrak{I},m}$. If $m = 0 \bmod N$, then $m = m_0 N$ for some m_0. If $m_0 = 0$, then $m = 0$ and $d \in \overline{\mathsf{Zero}}_k^{\mathfrak{I},m}$ by (44). If $m_0 > 0$, we distinguish the following two cases.

– If $m_0 = 0 \bmod 2^N$, then, by Claim 1, $2^N - 1 \in \mathsf{val}(m_0-1)$. Claim 2 then yields $d \in I_k^{\mathfrak{I},m_0N-1}$. By (45), $d \in \mathsf{Zero}_k^{\mathfrak{I},m_0N}$ (recall $m_0N = m$).

– If $m_0 \neq 0 \bmod 2^N$, we denote with $p_0 = (m_0-1) \bmod 2^N$. By Claim 1, there is $p_0 \in \mathsf{val}(m_0 - 1)$. Since $p_0 \neq 2^N-1$, Claim 2 yields $d \in \overline{I}_k^{\mathfrak{I},m_0N+1}$. By (46), we obtain $d \in \overline{\mathsf{Zero}}_k^{\mathfrak{I},m_0N}$ (recall $m_0N = m$).

This finishes the proof of Lemma 13. ❏

For the second step, take a deterministic Turing machine $\mathfrak{A} = (Q, \Sigma, \Gamma, q_0, \delta, F)$ with a k-EXPSPACE-hard word problem, that is, there is a polynomial $p(n)$ such that $\mathfrak{A}$ needs on any input of length n at most $\exp_k(p(n))$ tape cells. A configuration of $\mathfrak{A}$ is represented along the temporal dimension, the computation of $\mathfrak{A}$ is a sequence of configurations; note the similarity to the above. The main difficulty is to communicate corresponding tape cells in two consecutive configurations; it is routine to do so using the TBox $\mathcal{T}_{k,p(n)}$ for inputs of length n (note that the size of $\mathcal{T}_{k,p(n)}$ is polynomial in k, n). We represent an input word $w = \sigma_1 \cdots \sigma_n$ by means of the ABox $\mathcal{A}_w$ defined as

$$\{\mathsf{Init}_k(a, 0), A_{\sigma_1}(a, 0), \dots, A_{\sigma_n}(a, n-1), A_{\not{b}}(a, n)\},$$

where A_σ, for $\sigma \in \Gamma$, are concept names that represent the tape alphabet. The signature further includes concept names Z_q, $q \in Q$ for representing the states of $\mathfrak{A}$. Acceptance is detected by an additional concept name Acc together with the concept inclusions

$$\begin{aligned} Q_q &\sqsubseteq \mathsf{Acc}, \quad \text{for each } q \in F,\\ \bigcirc_F \mathsf{Acc} &\sqsubseteq \mathsf{Acc},\\ \mathsf{Init}_k \sqcap \mathsf{Acc} &\sqsubseteq B. \end{aligned}$$

It is then not hard to show that the sketched TBox $\mathcal{T}$ satisfies the following:

$$(\mathcal{T}, \mathcal{A}_w) \models B(a, 0) \quad \text{iff} \quad \mathfrak{A} \text{ accepts } w.$$

This completes the proof of Theorem 2. ❏

4 Foundations of Query Answering in $\mathcal{TEL}^\bigcirc$

In the following lemma we show how to establish the normal form.

Lemma 14 *For every $\mathcal{TEL}^\bigcirc$-TBox $\mathcal{T}$ with concept names* CN*, we can construct in polynomial time a $\mathcal{TEL}^\bigcirc$-TBox $\mathcal{T}'$ with concept names $\mathsf{CN}' \supseteq \mathsf{CN}$ such that $\mathcal{T}'$ is a model conservative extension of $\mathcal{T}$:*

– *every model of $\mathcal{T}'$ is a model of $\mathcal{T}$ and*
– *every model of $\mathcal{T}$ can be extended to a model of $\mathcal{T}'$ by giving a suitable interpretation to fresh concept names $\mathsf{CN}' \setminus \mathsf{CN}$.*

Proof. Let $\mathsf{sub}(\mathcal{T})$ be the set of sub-concepts appearing in $\mathcal{T}$. For every $C \in \mathsf{sub}(\mathcal{T})$, take a fresh concept name X_C and define concept $\overline{C}$ as follows:

$$\overline{C} = \begin{cases} A, & \text{if } C = A \in \mathsf{CN},\\ \exists r.X_D, & \text{if } C = \exists r.D,\\ \bigcirc_* X_D, & \text{if } C = \bigcirc_* D,\\ X_{D_1} \sqcap X_{D_2}, & \text{if } C = D_1 \sqcap D_2. \end{cases}$$

Now, let $\mathcal{T}'$ consist of the following concept inclusions:

$$\begin{aligned} &\overline{C} \sqsubseteq X_D, && \text{for all } C \sqsubseteq D \in \mathcal{T};\\ &\overline{C} \sqsubseteq X_C, X_C \sqsubseteq \overline{C}, && \text{for all } C \in \mathsf{sub}(\mathcal{T}). \end{aligned}$$

Observe that $\mathcal{T}'$ is almost of the required shape: conjunctions on the right-hand side are just abbreviations, and, for instance, $\bigcirc_P$ on the right-hand side can be expressed via $\bigcirc_F$ on the left-hand side. Finally, it is routine to prove the two properties mentioned above. ❏

Theorem 4 *Let $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ be the canonical quasimodel of $(\mathcal{T}, \mathcal{A})$ with a $\mathcal{TEL}^\bigcirc$-TBox $\mathcal{T}$ and $D = \mathsf{ind}(\mathcal{A}) \cup \mathsf{CN}$. Then, for every $A \in \mathsf{CN}$, $a \in \mathsf{ind}(\mathcal{A})$, $i \in \mathbb{Z}$, we have*

$$(\mathcal{T}, \mathcal{A}) \models A(a, i) \quad \textit{iff} \quad A \in \pi_a(i).$$

Proof. ($\Leftarrow$) Suppose that $(\mathcal{T}, \mathcal{A}) \not\models A(a, i)$, that is, there is a model $\mathfrak{I} = (\Delta^\mathfrak{I}, (\mathcal{I}_n)_{n\in\mathbb{Z}})$ of $(\mathcal{T}, \mathcal{A})$ such that $a \notin A^{\mathfrak{I},i}$. We can read off a collection of traces from $\mathfrak{I}$ as follows. First assign a trace π_a to every $a \in \Delta^\mathfrak{I}$ by taking:

$$\pi_a(i) = \{C \in \mathsf{CN} \mid a \in C^{\mathfrak{I},i}\}, \qquad \text{for every } i \in \mathbb{Z}.$$

The definition of π_B, $B \in \mathsf{CN}$ is slightly more complicated. Intuitively, $\pi_B(i)$ contains a concept name C iff every instance of B, say at time point n, in $\mathfrak{I}$ evolves to an instance of C at time point $n + i$. Formally, we put for $i \in \mathbb{Z}$, $A \in \mathsf{CN}$:

$$\pi_B(i) = \{C \in \mathsf{CN} \mid \mathfrak{I} \models B \sqsubseteq \bigcirc^i C\},$$

where $\bigcirc^i$ is $\bigcirc_F^i$ if $i \geq 0$ and $\bigcirc_P^{-i}$ if $i < 0$. It remains to verify that the collection $\{\pi_d \mid d \in D\}$ satisfies properties **(t1)–(t2)**, and **(q1)–(q4)**. Properties **(t1)**, **(t2)**, and **(q3)** are satisfied since $\mathfrak{I} \models \mathcal{T}$. Properties **(q1)** and **(q2)** are satisfied since $\mathfrak{I} \models \mathcal{A}$. For verifying **(q4)**, let $C \in \pi_d(n)$, for some $d \in D$, $n \in \mathbb{Z}$ and $C \sqsubseteq \exists r.B \in \mathcal{T}$ for some rigid role $r \in \mathsf{N}_\mathsf{R}^\mathsf{rig}$; the case $r \in \mathsf{N}_\mathsf{R}^\mathsf{loc}$ is similar. Moreover, suppose that $E \in \mathsf{proj}_r(\pi_B)(i - n)$ for some $i \in \mathbb{Z}$, $E \in \mathsf{CN}$. We need to show that $E \in \pi_d(i)$.

By the definition of proj_r, there is $E' \in \pi_B(i - n)$ such that $\exists r.E' \sqsubseteq E \in \mathcal{T}$. We distinguish cases on d.

– If $d \in \mathsf{ind}(\mathcal{A})$, then $C \in \pi_d(n)$ implies $d \in C^{\mathfrak{I},n}$. Since $\mathfrak{I} \models \mathcal{T}$ and $C \sqsubseteq \exists r.B \in \mathcal{T}$, there is some $d' \in B^{\mathfrak{I},n}$ such that $(d, d') \in r^{\mathfrak{I},n}$. As $E' \in \pi_B(i - n)$, we know that $\mathfrak{I} \models B \sqsubseteq \bigcirc^{i-n} E'$, and thus $d' \in E'^{\mathfrak{I},i}$. As $\exists r.E' \sqsubseteq E \in \mathcal{T}$, we obtain $d \in E^{\mathfrak{I},i}$, whence $E \in \pi_d(i)$.

– If $d \in \mathsf{CN}$, then $C \in \pi_d(n)$ implies $\mathfrak{I} \models d \sqsubseteq \bigcirc^n C$. Since $\mathfrak{I} \models \mathcal{T}$ and $C \sqsubseteq \exists r.B \in \mathcal{T}$, we have $\mathfrak{I} \models d \sqsubseteq \bigcirc^n \exists r.B$. From $E' \in \pi_B(i - n)$, we obtain $\mathfrak{I} \models B \sqsubseteq \bigcirc^{i-n} E'$, and thus, $\mathfrak{I} \models d \sqsubseteq \bigcirc^n \exists r.\bigcirc^{i-n} E'$. As $\exists r.E' \sqsubseteq E \in \mathcal{T}$, we obtain $\mathfrak{I} \models x \sqsubseteq \bigcirc^i E$, whence $E \in \pi_d(i)$.

By the initial assumption $a \notin A^{\mathfrak{I},i}$ and the definition of π_a, we have $A \notin \pi_a(i)$. Thus, $A \notin \pi_a(i)$ in the canonical quasimodel, which finishes the proof of the "$\Leftarrow$"-direction.

($\Rightarrow$) Let $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ be the canonical quasimodel for $(\mathcal{T}, \mathcal{A})$. We define the *unravelling of* $\mathfrak{Q}$ inductively as follows. We start with an interpretation $\mathfrak{I}_0 = (\Delta_0, (\mathcal{I}_{0n})_{n\in\mathbb{Z}})$ defined by taking:

$$\Delta_0 = \mathsf{ind}(\mathcal{A}),$$

$$
\begin{aligned}
A^{\mathfrak{I}_0,n} &= \{a \in \mathsf{ind}(\mathcal{A}) \mid A \in \pi_a(n)\},\\
r^{\mathfrak{I}_0,n} &= \{(a,b) \mid r(a,b,n) \in \mathcal{A}, r \in \mathsf{N}_\mathsf{R}^{\mathsf{loc}}\} \cup\\
&\qquad \{(a,b) \mid r(a,b,m) \in \mathcal{A}, r \in \mathsf{N}_\mathsf{R}^{\mathsf{rig}}\}.
\end{aligned}
$$

The interpretation $\mathfrak{I}_{i+1}$ is obtained from $\mathfrak{I}_i$ by applying the following rule where applicable:

- if $a \in A^{\mathfrak{I}_i,n}$ and $A \sqsubseteq \exists r.B \in \mathcal{T}$, then extend Δ_i with a fresh element b and
  - if $r \in \mathsf{N}_\mathsf{R}^{\mathsf{loc}}$, then put $(a,b) \in r^{\mathfrak{I}_{i+1},n}$;
  - if $r \in \mathsf{N}_\mathsf{R}^{\mathsf{rig}}$, then put $(a,b) \in r^{\mathfrak{I}_{i+1},m}$ for all $m \in \mathbb{Z}$;
  - for all $E \in \mathsf{CN}$, $m \in \mathbb{Z}$, put
  $$b \in E^{\mathfrak{I}_{i+1},n+m} \quad \text{iff} \quad E \in \pi_B(m).$$

Obtain $\mathfrak{I}$ from $(\mathfrak{I}_i)_{i\geq 0}$ by taking for all $n \in \mathbb{Z}$, concept names A, and role names r:

$$\Delta = \bigcup_{i\geq 0} \Delta_i, \quad A^{\mathfrak{I},n} = \bigcup_{i\geq 0} A^{\mathfrak{I}_i,n}, \quad r^{\mathfrak{I},n} = \bigcup_{i\geq 0} r^{\mathfrak{I}_i,n}.$$

Let us verify that $\mathfrak{I} \models (\mathcal{T},\mathcal{A})$. The definition of $\mathfrak{I}_0$ together with properties **(q1)** and **(q2)** yields $\mathfrak{I} \models \mathcal{A}$. From properties **(t1)** and **(t2)** together with the definition of $A^{\mathfrak{I},n}$, we obtain that $\mathfrak{I} \models \alpha$ for all $\alpha \in \mathcal{T}$ of the form $A_1 \sqcap A_2 \sqsubseteq A$, and $\bigcirc_* A_1 \sqsubseteq A$. For the remaining concept inclusions from $\mathcal{T}$, we provide more detail. Let $A \sqsubseteq \exists r.B \in \mathcal{T}$ and $a \in A^{\mathfrak{I},n}$ for some $a \in \Delta$. By the above rule, we have that the freshly introduced element b satisfies $(a,b) \in r^{\mathfrak{I},n}$ and $b \in B^{\mathfrak{I},n}$.

Consider finally a concept inclusion $\exists r.B \sqsubseteq A \in \mathcal{T}$, domain elements $a,b \in \Delta^{\mathfrak{I}}$ satisfying $(a,b) \in r^{\mathfrak{I},n}$ and $b \in B^{\mathfrak{I},n}$. We need to show that $a \in A^{\mathfrak{I},n}$. If $(a,b) \in r^{\mathfrak{I},n}$ because $(a,b) \in r^{\mathfrak{I}_0,n}$, then $a \in A^{\mathfrak{I},n}$ due to the definition of interpretation of concept names and **(q3)**. If (a,b) was added to $r^{\mathfrak{I}_i,n+1}$ in some rule application, we obtain $a \in A^{\mathfrak{I},n}$ due to the definition of interpretation of concept names and **(q4)**.

We conclude the proof by noting that if $A \notin \pi_a(i)$, for some $A \in \mathsf{CN}$, $a \in \mathsf{ind}(\mathcal{A})$, $i \in \mathbb{Z}$, then, by definition of $\mathfrak{I}$, $a \notin A^{\mathfrak{I},i}$ and hence $(\mathcal{T},\mathcal{A}) \not\models A(a,i)$. ❑

Theorem 5 *TAQ answering over ultimately periodic $\mathcal{TEL}^{\bigcirc}$-TBoxes is decidable and* PSPACE*-complete in data complexity.*

Proof. The lower bound is by Lemma 12. The matching upper bound is established in Lemma 15 below for the rewriting $\Pi_\mathcal{T}$ constructed in Section 4. ❑

Lemma 15 *For all ABoxes $\mathcal{A}$, $A \in \mathsf{CN}$, $a \in \mathsf{ind}(\mathcal{A})$, $i \in \mathbb{Z}$, we have $(\Pi_\mathcal{T},\mathcal{A}) \models A(a,i)$ iff $(\mathcal{T},\mathcal{A}) \models A(a,i)$.*

Proof. ($\Leftarrow$) Suppose that $(\Pi_\mathcal{T},\mathcal{A}) \not\models A(a,i)$, that is, $A(a,i) \notin \Pi_\mathcal{T}(\mathcal{A})$, where $\Pi_\mathcal{T}(\mathcal{A})$ is the minimal Herbrand model of $(\Pi_\mathcal{T},\mathcal{A})$. We read off a collection of traces $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ precisely as in the "$\Leftarrow$"-direction in the proof of Theorem 4, that is, for every $i \in \mathbb{Z}$, $a \in \mathsf{ind}(\mathcal{A})$, $B \in \mathsf{CN}$:

$$
\begin{aligned}
\pi_a(i) &= \{C \in \mathsf{CN} \mid C(a,i) \in \Pi_\mathcal{T}(\mathcal{A})\},\\
\pi_B(i) &= \{C \in \mathsf{CN} \mid \forall a,n : B(a,n) \in \Pi_\mathcal{T}(\mathcal{A})\\
&\qquad\qquad \Rightarrow C(a,n+i) \in \Pi_\mathcal{T}(\mathcal{A})\}.
\end{aligned}
$$

It is routine to verify that $\mathfrak{Q}$ satisfies **(t1)**, **(t2)**, and **(q1)**–**(q4)**.

It remains to show that $\mathfrak{Q}$ satisfies the quasimodel conditions. **(t1)** is satisfied by formulas (4). For **(t2)**, assume $A \in \pi_d(n)$ and $\bigcirc_P A \sqsubseteq B \in \mathcal{T}$. The latter and **(t2)** implies that $B \in \hat{\pi}_A(1)$, where $\{\hat{\pi}_d \mid d \in D\}$ are the traces in the canonical quasimodel. By construction of $\Pi_\mathcal{T}$, we have a rule $B(x,t+1) \leftarrow A(x,t)$. We distinguish cases on d.

- if $d \in \mathsf{ind}(\mathcal{A})$, we get $A(d,i) \in \Pi_\mathcal{T}(\mathcal{A})$. By the mentioned rule, also $B(d,i+1) \in \Pi_\mathcal{T}(\mathcal{A})$ and hence $B \in \pi_d(i+1)$;
- if $d = X \in \mathsf{CN}$, then for all b,n, if $X(b,n) \in \Pi_\mathcal{T}(\mathcal{A})$, then $A(b,i+n) \in \Pi_\mathcal{T}(\mathcal{A})$. By the above rule, $X(b,n) \in \Pi_\mathcal{T}(\mathcal{A})$ also implies $B(b,i+n+1) \in \Pi_\mathcal{T}(\mathcal{A})$ for all b,n. Hence, $B \in \pi_d(i+1)$.

Properties **(q1)** and **(q2)** are clear as $\mathcal{A} \in \Pi_\mathcal{T}(\mathcal{A})$ and **(q3)** follows directly from formulas (3) and (5) included in $\Pi_\mathcal{T}$. For **(q4)**, assume that $A \in \pi_d(n)$, $A \sqsubseteq \exists r.B \in \mathcal{T}$, and $E \in \mathsf{proj}_r(\pi_B)(i-n)$ for some $i \in \mathbb{Z}$. We need to show that $E \in \pi_d(i)$. By definition of proj_r, there is some $E' \in \pi_B(i-n)$ such that $\exists r.E' \sqsubseteq E \in \mathcal{T}$. Since $A \sqsubseteq \exists r.B \in \mathcal{T}$ and **(q4)** applied to the canonical quasimodel, we get $\mathsf{proj}_r(\pi_B) \subseteq^0 \pi_A$. In particular, $E \in \pi_A(j)$ whenever $E' \in \pi_B$ and thus $E \in \pi_A(i-n)$. The construction of $\Pi_\mathcal{T}$, in particular, the rules included for π_A, implies the rule $E(x,t+i-n) \leftarrow A(x,t)$. We distinguish cases on d.

- If $d \in \mathsf{ind}(\mathcal{A})$, then $A(d,n) \in \Pi_\mathcal{T}(\mathcal{A})$. As a consequence of the aforementioned rule, we have $E(d,i) \in \Pi_\mathcal{T}(\mathcal{A})$; whence $E \in \pi_d(i)$.
- If $d = X \in \mathsf{CN}$, then for all b,m, if $X(b,m) \in \Pi_\mathcal{T}(\mathcal{A})$, then $A(b,n+m) \in \Pi_\mathcal{T}(\mathcal{A})$. By the above rule, $X(b,m) \in \Pi_\mathcal{T}(\mathcal{A})$ also implies $E(b,m+i) \in \Pi_\mathcal{T}(\mathcal{A})$, for all b,m. Hence, $B \in \pi_d(i)$.

Since $A(a,i) \notin \Pi_\mathcal{T}(\mathcal{A})$, we have $A \notin \pi_a(i)$; thus, $A \notin \hat{\pi}_a(i)$ in the canonical quasimodel. Theorem 4 then yields $(\mathcal{T},\mathcal{A}) \not\models A(a,i)$.

($\Rightarrow$) Assume that $(\mathcal{T},\mathcal{A}) \not\models A(a,i)$ and thus $A \notin \pi_a(i)$ in the canonical quasimodel $\{\pi_d \mid d \in D\}$. We show that $A(a,i) \notin \Pi_\mathcal{T}(\mathcal{A})$ by noting that the set M, defined as

$$
\begin{aligned}
M \;=\; &\{B(b,n) \mid B \in \pi_b(n), b \in \mathsf{Ind}(\mathcal{A})\} \cup\\
&\{r(a,b,n) \mid r(a,b,m) \in \mathcal{A}, r \in \mathsf{N}_\mathsf{R}^{\mathsf{rig}}\} \cup\\
&\{r(a,b,n) \mid r(a,b,n) \in \mathcal{A}, r \in \mathsf{N}_\mathsf{R}^{\mathsf{loc}}\},
\end{aligned}
$$

clearly is a model for $\Pi_\mathcal{T}$ and $\mathcal{A}$, thus $M \supseteq \Pi_\mathcal{T}(\mathcal{A})$, but $A(a,i) \notin M$. ❑

5 Restricted Use of Rigid Roles

Theorem 6 *TAQ answering over $\mathcal{TEL}^{\bigcirc}_{\mathsf{loc}}$ is* PSPACE*-complete in combined and* PTIME*-complete in data complexity.*

Proof. PTIME-hardness in data complexity follows from PTIME-hardness for atomic query answering in $\mathcal{EL}$. PSPACE-hardness in combined complexity follows from the (proof of) PSPACE-hardness for entailment in Horn-LTL [Chen and Lin, 1993], which, in fact, shows PSPACE-hardness of checking whether $\Box\varphi \models p \rightarrow q$, where φ is a conjunction of implications involving only $\bigcirc_*$ and $\wedge$, and p and q are propositional variables. Now, $\Box\varphi$ can be translated in a straightforward way into a TBox $\mathcal{T}_\varphi$ (using only concept and no role names).

For the upper bounds, let $\varphi_{\mathcal{T},\mathcal{A}}$ be the Horn-LTL formula constructed in the main part of the paper, and let $\mathfrak{M}$ be its minimal model. We prove correctness of $\varphi_{\mathcal{T},\mathcal{A}}$ in the following sense:

Claim. For all $a \in \mathsf{ind}(\mathcal{A})$, $i \in \mathsf{tem}(\mathcal{A})$, $A \in \mathsf{CN}$, we have $(\mathfrak{M}, i) \models P_{A,a}$ iff $(\mathcal{T}, \mathcal{A}) \models A(a, i)$.

Proof of Claim. ($\Rightarrow$) Suppose $(\mathcal{T}, \mathcal{A}) \not\models A(a, i)$. By Theorem 4, we have $A \notin \pi_a(i)$. Define a temporal LTL interpretation $\mathfrak{M}'$ by taking, for all $i \in \mathbb{Z}$, $d \in D$, $B \in \mathsf{CN}$,

$$(\mathfrak{M}', i) \models P_{B,d} \quad \text{iff} \quad B \in \pi_d(i).$$

It is routine to prove that $\mathfrak{M}'$ is a model of $\varphi_{\mathcal{T},\mathcal{A}}$; note that the formulas in that order correspond almost literally to **(t1)–(t2)** and **(q1)–(q4)**. By construction, we have $(\mathfrak{M}', i) \not\models P_{A,a}$ and thus, by the minimality of $\mathfrak{M}$, we obtain $(\mathfrak{M}, i) \not\models P_{A,a}$.

($\Leftarrow$) Suppose $(\mathfrak{M}, i) \not\models P_{A,a}$. We define a quasimodel $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ with $A \notin \pi_a(i)$. Theorem 4 will then yield $(\mathcal{T}, \mathcal{A}) \not\models A(a, i)$. Define $\mathfrak{Q}$ by taking

$$\pi_d(i) = \{B \in \mathsf{CN} \mid (\mathfrak{M}, i) \models P_{B,d}\} \quad \text{for all } i \in \mathbb{Z}, d \in D.$$

It is routine to verify that $\mathfrak{Q}$ is indeed a quasimodel. This finishes the proof of Claim.

Theorem 6 follows from Claim by the runtime arguments given in the paper. ❑

Theorem 7 *TAQ answering over $\mathcal{TEL}^{\circ}_{\mathsf{l\text{-}rig}}$ is* PSPACE-*complete in data complexity and in* EXPTIME *in combined complexity.*

Proof. PSPACE-hardness in data complexity is by Theorem 2. For the upper bound, let $\Pi'_{\mathcal{T}}$ be the DATALOG$_{1S}$-program constructed as in the proof of Theorem 5 in Section 4 but where the rules for traces π_B in $\Pi_{\mathcal{T}}$ are replaced by the set of rules

$$A(x, t) \leftarrow B(x, t), \text{ for all } A \in \pi_B(0).$$

It is routine to verify that $\Pi'_{\mathcal{T}}$ can be used for TAQ answering in the sense of Lemma 15. For showing the complexity upper bounds, denote with $\mathcal{T}' \subseteq \mathcal{T}$ the subset of $\mathcal{T}$ not mentioning rigid roles, and let $\{\pi'_d \mid d \in D\}$ be the canonical quasimodel for $(\mathcal{T}', \emptyset)$. It should be clear that, by the restriction posed on the occurrence of rigid roles, we have $A \in \pi_B(0)$ iff $A \in \pi'_B(0)$. The latter can be decided in PTIME data and PSPACE combined complexity, see Theorem 6. This yields the result since query answering in DATALOG$_{1S}$ is in PSPACE data and EXPTIME combined complexity. ❑

6 Acyclicity Conditions

Theorem 8 *TAQ answering over acyclic $\mathcal{TEL}^{\circ}$ is in* LOGTIME-*uniform AC^0 in data complexity and in* PTIME *in combined complexity.*

Proof. We start with data complexity. It suffices to show that $\varphi_{\mathcal{T},A}(x, t)$ is indeed an FO(+)-rewriting of $\mathcal{T}, A(x, t)$. Denote with $\mathfrak{I}_{\mathcal{A}}$ the ABox $\mathcal{A}$ viewed as an interpretation.

Claim. For all temporal ABoxes $\mathcal{A}$, $A \in \mathsf{CN}$, $a \in \mathsf{ind}(\mathcal{A})$, and $i \in \mathsf{tem}(\mathcal{A})$, $(\mathcal{T}, \mathcal{A}) \models A(a, i)$ iff $\mathfrak{I}_{\mathcal{A}} \models \varphi_{\mathcal{T},A}(a, i)$.

Proof of the Claim. The proof is by induction on the 'depth' of the expansion of A relative to the acyclic TBox $\mathcal{T}$. If A is primitive, that is, has depth 0, the statement is direct. If $A \equiv B_1 \sqcap B_2 \in \mathcal{T}$, the statement follows directly from the induction hypothesis. For the other cases, the proof uses the *universal model* $\mathfrak{I}_{\mathcal{T},\mathcal{A}}$ of $(\mathcal{T}, \mathcal{A})$, which is the result of unravelling the canonical quasimodel, and which satisfies the following property for all $a \in \mathsf{ind}(\mathcal{A})$ and $i \in \mathbb{Z}$:

$$(\mathcal{T}, \mathcal{A}) \models A(a, i) \quad \text{iff} \quad \mathfrak{I}_{\mathcal{T},\mathcal{A}} \models A(a, i). \tag{50}$$

Assume first $A \equiv \exists r.B \in \mathcal{T}$, $r \in \mathsf{N}_\mathsf{R}^\mathsf{rig}$ a rigid role, and $(\mathcal{T}, \mathcal{A}) \not\models A(a, i)$, and, by (50), also $\mathfrak{I}_{\mathcal{T},\mathcal{A}} \not\models A(a, i)$. Moreover, also by (50), we obtain $\mathfrak{I}_{\mathcal{T},\mathcal{A}} \not\models D(a, i)$ for all concept names D with $\mathcal{T} \models D \sqsubseteq A$. Assume, for the sake of contradiction, that $\mathfrak{I}_{\mathcal{T},\mathcal{A}} \models \exists y t' \, (r(a, y, t') \wedge \varphi_{\mathcal{T},B}(y, i))$. By the induction hypothesis, $\mathfrak{I}_{\mathcal{T},\mathcal{A}} \models \exists y t' \, (r(a, y, t') \wedge B(y, i))$, and hence $\mathfrak{I}_{\mathcal{T},\mathcal{A}} \models (\exists r.B)(a, i)$, a contradiction. Thus, $\mathfrak{I}_{\mathcal{T},\mathcal{A}} \not\models \varphi_{\mathcal{T},A}(a, i)$, whence $\mathfrak{I}_{\mathcal{A}} \not\models \varphi_{\mathcal{T},A}(a, i)$.

For the converse direction, assume that $(\mathcal{T}, \mathcal{A}) \models A(a, i)$ and thus $\mathfrak{I}_{\mathcal{T},\mathcal{A}} \models A(a, i)$. It is straightforward to show that this is the case iff there is some $D(a, i) \in \mathcal{A}$ such that $\mathcal{T} \models D \sqsubseteq A$ or there are $b \in \mathsf{ind}(\mathcal{A})$, $j \in \mathsf{tem}(\mathcal{A})$ such that $r(a, b, j) \in \mathcal{A}$ and $\mathfrak{I}_{\mathcal{A}} \models B(b, i)$, that is, $(\mathcal{T}, \mathcal{A}) \models B(b, i)$, which, by the induction hypothesis, implies $\mathfrak{I}_{\mathcal{A}} \models \varphi_{\mathcal{T},B}(b, i)$. In the former case, we obtain $\mathfrak{I}_{\mathcal{A}} \models \varphi_{\mathcal{T},A}(a, i)$ as there is a disjunct for D. In the latter case, we have $\mathfrak{I}_{\mathcal{A}} \models \exists y t' \, (r(a, y, t') \wedge \varphi_{\mathcal{T},B}(y, i))$ and hence $\mathfrak{I}_{\mathcal{A}} \models \varphi_{\mathcal{T},A}(a, i)$.

The case when $r \in \mathsf{N}_\mathsf{R}^\mathsf{loc}$ is a local role is almost identical (with the quantifier $\exists t'$ omitted and t replaces t'). The case $A \equiv \bigcirc_* B \in \mathcal{T}$ can be dealt with in the same way. This finishes the proof of the claim and thus shows that $\varphi_{\mathcal{T},A}$ is indeed an FO(+)-rewriting over the infinite models of the form $\mathfrak{I}_{\mathcal{A}}$.

For PTIME combined complexity it suffices to note that, by acyclicity of $\mathcal{T}$, the traces in the canonical quasimodel $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ have very restricted temporal extension. In particular, it is not hard to verify, see Lemma 17 in the proof of Theorem 10, that for all $a \in \mathsf{ind}(\mathcal{A})$, $B \in \mathsf{CN}$ and $j \in \mathbb{Z}$:

$$\pi_a(j) = \emptyset, \quad \text{if } j < \min \mathcal{A} - |\mathcal{T}| \text{ or } j > \max \mathcal{A} + |\mathcal{T}|;$$
$$\pi_B(j) = \emptyset, \quad \text{if } j < -|\mathcal{T}| \text{ or } j > |\mathcal{T}|.$$

Thus, the algorithm for constructing the canonical quasimodel terminates in polynomial time. ❑

DL-acyclicity

Theorem 9 *TAQ answering over DL-acyclic $\mathcal{TEL}^{\circ}$-TBoxes of depth $k \geq 1$ is k-*EXPSPACE*-complete in combined complexity and NC^1-complete in data complexity.*

The following Lemma shows how to characterize traces in the canonical quasimodel by Horn-LTL formulas. The applied technique will be useful in the proof of Theorem 9. We define function $\exp^k(n)$ by taking

$$\exp^0(n) = n, \qquad \exp^{k+1}(n) = 2^{\exp^k(n)}$$

(with the superscript k rather than subscript in Theorem 2).

Lemma 16 *For all $k \geq 0$ and $B \in \mathsf{CN}$ with $\ell_{\text{DL}}(B) = k$, there is a propositional Horn-LTL formula φ_B of size $\exp^k(|\mathcal{T}|)$ over variables P_A, $A \in \mathsf{CN}$, such that, for all $n \in \mathbb{Z}$,*

$$A \in \pi_B(n) \quad \textit{iff} \quad \varphi_B \models \bigcirc^n P_A. \tag{51}$$

Proof. We prove it by induction on k. Let φ_0 be a conjunction of $\Box(P_{A_1} \wedge P_{A_2} \rightarrow P_A)$, for $A_1 \sqcap A_2 \sqsubseteq A \in \mathcal{T}$, and $\Box(\bigcirc_* P_B \rightarrow P_A)$, for $\bigcirc_* B \sqsubseteq A \in \mathcal{T}$.

For the basis of induction, $\ell_{\text{DL}}(B) = 0$, observe that, by definition of ℓ_{DL} and **(q4)**, only CIs without existential restrictions are relevant for the construction of π_B. Taking $\varphi_B = B \wedge \varphi_0$ clearly satisfies (51).

For the induction step, take B with $\ell_{\text{DL}}(B) > 0$ and let $B_1, \dots, B_k$ be the set of all concept names such that

$$A_i \sqsubseteq \exists r.B_i \in \mathcal{T} \text{ and } A_i \in \pi_B(n), \text{ for some } n \in \mathbb{Z}.$$

Since $\ell_{\text{DL}}(B_i) < \ell_{\text{DL}}(B)$, by the induction hypothesis, there are formulas φ_{B_i} satisfying (51). Starting from φ_{B_i}, it is not hard to construct formulas ψ_{B_i} that describe $\mathsf{proj}_r(\pi_{B_i})$ (in the sense of (51)). Note that $\mathsf{proj}_r(\pi_{B_i})$ is ultimately p-periodic for some $p \leq \exp^{\ell_{\text{DL}}(B)}(|\mathcal{T}|)$ and let m_F, m_P, p_F, p_P be the respective constants. From ψ_{B_i}, we obtain a conjunction χ_{B_i} in the "loop-normal form": $\bigcirc_F^j P_C$ for all $0 \leq j < m_F$ with $\psi_{B_i} \models \bigcirc_F^j P_C$; $\bigcirc_F^{m_F} F_i$; $\Box(F_i \rightarrow \bigcirc_F^{p_F} F_i)$; $\Box(F_i \rightarrow \bigcirc_F^j C)$ for all $0 \leq j < p_F$ with $\psi_{B_i} \models \bigcirc_F^{m_F+j} P_C$; and symmetric formulas for m_P, p_P; cf. the DATALOG$_{1S}$ program in the proof of Theorem 5. Note that this incurs an exponential blow-up. It is routine to show that $\varphi_B = B \wedge \varphi_0 \wedge \Box \bigwedge_i (A_i \rightarrow \chi_{B_i})$ satisfies (51). ❑

We continue now with the proof of Theorem 9.

Proof. A closer inspection of the non-elementary lower bound for $\mathcal{TEL}^\circ$ in Theorem 2 reveals that the TBoxes used are DL-acyclic. In fact, it is shown that subsumption, and so TAQ answering, over TBoxes of depth k is k-EXPSPACE-hard. NC^1-hardness in data complexity follows from the fact that the word problem of NFAs can be reduced to TAQ answering, even without the DL dimension, see [Artale et al., 2015, Theorem 9].

In order to show the matching upper bounds, consider a temporal knowledge base $(\mathcal{T}, \mathcal{A})$ with $\mathcal{T}$ of depth k. We devise a completion procedure for TAQ answering. For this purpose, we define sets $\mathcal{A}_i$ consisting of all assertions $A(a,i)$ and $r(a,b,i)$ in $\mathcal{A}$ and, for $r \in \mathsf{N}_\mathsf{R}^\mathsf{rig}$, assertions $r(a,b,i)$ such that $r(a,b,j) \in \mathcal{A}$ for some $j \in \mathbb{Z}$. The algorithm exhaustively adds assertions $A(a,i)$ to $\mathcal{A}$ if one of the following holds:

(a) $(\mathcal{T}, \mathcal{A}_i) \models A(a,i)$;

(b) $B(a, i \; \mathsf{op}_* \; 1) \in \mathcal{A}$ and $\bigcirc_* B \sqsubseteq A \in \mathcal{T}$.

Intuitively, Condition (b) captures the immediate temporal implications of the TBox, while Condition (a) takes into account the role structure of the individuals in $\mathcal{A}$. Soundness of the algorithm should be clear. For completeness, it is routine to construct the canonical quasimodel of $(\mathcal{T}, \mathcal{A})$ given the result of the algorithm. The crucial observation is that $\mathcal{A}_{\min \mathcal{A}}$ (respectively, $\mathcal{A}_{\max \mathcal{A}}$) determines the traces π_a, $a \in \mathsf{ind}(\mathcal{A})$, in the quasimodel on all $i < \min \mathcal{A}$ (respectively, $i > \max \mathcal{A}$).

It turns out that, in Condition (a), $\mathcal{A}_i$ can be replaced with a suitably defined *quotient* $\mathcal{B}_i$ *of* $\mathcal{A}_i$. Intuitively, the considered query language can only distinguish all possible distinct trees of depth k, whose number depends on $|\mathcal{T}|$ only; thus, the size of $\mathcal{B}_i$ is independent of the size of $\mathcal{A}$. For making the notion of "quotient" precise, assume we want to check whether $(\mathcal{T}, \mathcal{A}_i) \models A(a,i)$, for a concept A with $\ell_{\text{DL}}(A) = k$. Let $\mathcal{A}_a^u$ be the unravelling of $\mathcal{A}_i$ from point a up to depth k such that for all $0 \leq j \leq k$, and all b in distance j of a, we have $B(b,j) \in \mathcal{A}_a^u$ only if $\ell_{\text{DL}}(B) = k - j$.

Claim 1. $(\mathcal{T}, \mathcal{A}_i) \models A(a,i)$ iff $(\mathcal{T}, \mathcal{A}_a^u) \models A(a,i)$.

Next, let $\sim$ be the smallest equivalence relation on $\mathsf{ind}(\mathcal{A}_a^u)$ such that $b \sim b'$ implies

- $B(b,i) \in \mathcal{A}_a^u$ iff $B(b',i) \in \mathcal{A}_a^u$, for all B and $i \in \mathsf{tem}(\mathcal{A})$, and
- for each $c \in \mathsf{ind}(\mathcal{A})$ and $i \in \mathsf{tem}(\mathcal{A})$ with $r(b,c,i) \in \mathcal{A}_a^u$, there is $c' \in \mathsf{ind}(\mathcal{A})$ with $r(b',c',i) \in \mathcal{A}_a^u$ and $c \sim c'$, and vice versa.

Denote by $[b]$ the equivalence class of b with respect to $\sim$ and let $\mathcal{B}_i$ consist of the following assertions:

$$A([b], i), \quad \text{for } A(b,i) \in \mathcal{A}_a^u,$$
$$r([b],[b'],i), \quad \text{for } r(b,b',i) \in \mathcal{A}_a^u.$$

Clearly, $\mathsf{ind}(\mathcal{B}_i) = \{[b] \mid b \in \mathsf{ind}(\mathcal{A}_a^u)\}$.

Claim 2. $(\mathcal{T}, \mathcal{A}_a^u) \models A(a,i)$ iff $(\mathcal{T}, \mathcal{B}_i) \models A(a,i)$.

Following the steps in the proof of Lemma 16, but taking account of the structure of $\mathcal{B}_i$, we can construct a formula $\varphi_{a,i}$ of size $\exp^k(|\mathcal{T}|)$ such that $\varphi_{a,i} \models P_A$ iff $(\mathcal{T}, \mathcal{B}_i) \models A(a,i)$ for all $A \in \mathsf{CN}$ with $\ell_{\text{DL}}(A) = k$. The former can be checked in space polynomial in the size of $\varphi_{a,i}$, which together with a polynomial-time algorithm for checking Condition (b), yields the k-EXPSPACE upper combined complexity bound.

Note that the implementation sketched for the k-EXPSPACE upper combined complexity bound does not yield an NC^1 upper *data* complexity bound. We explain now how Conditions (a) and (b) can be implemented in NC^1. We begin with Condition (a) and give an algorithm that determines $\mathcal{B}_i$ "on-the fly", which requires some notation. For $0 \leq j \leq k$, let

$$\mathsf{CN}_j = \{A \in \mathsf{CN} \mid \ell_{\text{DL}}(A) = k - j\}.$$

Define equivalence relations $\sim_j^i$, for all $0 \leq j \leq k$ and $i \in \mathsf{tem}(\mathcal{A})$, similarly to $\sim$ above:

- $b \sim_0^i b'$ iff
  - $B(b,i) \in \mathcal{A} \Leftrightarrow B(b',i) \in \mathcal{A}$, for all $B \in \mathsf{CN}_0$;
- for $j > 0$, we take $b \sim_j^i b'$ iff
  - $B(b,i) \in \mathcal{A} \Leftrightarrow B(b',i) \in \mathcal{A}$, for all $B \in \mathsf{CN}_j$,
  - for each $c \in \mathsf{ind}(\mathcal{A})$ with $r(b,c,i) \in \mathcal{A}$, there is $c' \in \mathsf{ind}(\mathcal{A})$ with $r(b',c',i) \in \mathcal{A}$ and $c \sim_{j-1}^i c'$, and vice versa.

Denote by $[a]_j^i$ the equivalence class of a with respect to $\sim_j^i$. Note that the maximum number of equivalence classes of $\sim_j^i$ depends only on $\mathcal{T}$ and is independent of $\mathcal{A}$. Moreover, each b, i, j give rise to a unique tree-shaped ABox $\mathcal{B}_{b,j}^i$ defined by induction on j as follows:

– for $j = 0$, let

$$\mathcal{B}^i_{b,0} = \{A([b]^i_0, i) \mid A(b,i) \in \mathcal{A}, A \in \mathsf{CN}_0\};$$

– for $j > 0$, let $\mathcal{B}^i_{b,j}$ be the union of

$$\{A([b]^i_j, i) \mid A(b,i) \in \mathcal{A}, A \in \mathsf{CN}_j\} \cup$$
$$\{r([b]^i_j, [b']^i_{j-1}, i) \mid r(b,b',i) \in \mathcal{A}\}$$

and all $\mathcal{B}^i_{b',j-1}$ such that $r(b,b',i) \in \mathcal{A}$.

We say that an equivalence class $[b]^i_j$ *implies* $B \in \mathsf{CN}_j$ if $(\mathcal{T}, \mathcal{B}^i_{b,j}) \models B(b,i)$. This condition will replace Condition (a); note that it does not depend on the size of $\mathcal{A}$.

Condition (b) cannot be directly implemented as we noted above because the number of steps it has to be applied on depends at least linearly on $\mathcal{A}$, not yielding an NC^1 algorithm. Instead, we implement it using Büchi automata on ω-words over alphabet $\Sigma = 2^{\mathsf{CN}}$. We say that an ω-word $\sigma_0\sigma_1\cdots$ is an *l-trace* (*r-trace*, respectively) for $\mathcal{T}$ if there is a trace π for $\mathcal{T}$ such that $\sigma_i = \pi(-i)$ ($\sigma_i = \pi(i)$, respectively) for all $i \geq 0$. Based on the formulas constructed in the proof of Lemma 16, one can construct Büchi automata $\mathfrak{A}_l$ and $\mathfrak{A}_r$ (whose size does not depend on $\mathcal{A}$) that accept precisely the l- and r-traces for $\mathcal{T}$ [Vardi, 2006]. Moreover, we can construct automata $\mathfrak{A}_B$, $B \in \mathsf{CN}$ with $L(\mathfrak{A}_B) = \{\sigma_0\sigma_1 \ldots \in \Sigma^\omega \mid B \in \sigma_0\}$.

For $a \in \mathsf{ind}(\mathcal{A})$ and $i \in \mathsf{tem}(\mathcal{A})$, define a finite word $w^r_{ai} = \sigma_0 \cdots \sigma_n$ by taking, for $0 \leq j \leq \max \mathcal{A} - i$,

$$B \in \sigma_j \quad \text{iff} \quad B(a, i+j) \in \mathcal{A}, \qquad \text{for all } B \in \mathsf{CN},$$

and, analogously but mirrored, a finite word w^l_{ai}. We say that a finite word $\sigma_1 \cdots \sigma_n$ is a *possible prefix* for $L \subseteq \Sigma^\omega$ if there is a word $\tau_1\tau_2 \cdots \in L$ with $\tau_i \supseteq \sigma_i$, for all $1 \leq i \leq n$.

The algorithm proceeds in $k+1$ rounds, $0 \leq j \leq k$ (starting in round $j = 0$) and maintains (polynomially many) processor units P_{abi} for $(a,b,i) \in \mathsf{ind}(\mathcal{A})^2 \times \mathsf{tem}(\mathcal{A})$. In each round j, the following steps are performed:

(i) label each $(a,i) \in \mathsf{ind}(\mathcal{A}) \times \mathsf{tem}(\mathcal{A})$ by its equivalence class with respect to $\sim^i_j$;

(ii) for each $B \in \mathsf{CN}_j$, add $B(a,i)$ to $\mathcal{A}$ if $[a]^i_j$ implies B;

(iii) for each $B \in \mathsf{CN}_j$, add $B(a,i)$ to $\mathcal{A}$ if either w^r_{ai} is not a possible prefix for $L(\mathfrak{A}_r \setminus \mathfrak{A}_B)$ or w^l_{ai} is not a possible prefix for $L(\mathfrak{A}_l \setminus \mathfrak{A}_B)$.

Intuitively, Step (ii) corresponds to Condition (a) while Step (iii) corresponds to Condition (b); Step (i) is a necessary preprocessing for (ii). Note that Steps (ii) and (iii) are meant to be exhaustive in the sense that they add $B \in \mathsf{CN}_j$ until nothing changes; this is fine because the number of tests depends only on CN and thus, is constant (independent of the size of $\mathcal{A}$). Completeness of Step (iii) essentially relies on the following:

Claim 3. If $A(a,i)$ is added to $\mathcal{A}$ via a sequence of applications of (b) then $A(a,i)$ is added to $\mathcal{A}$ via a sequence of steps (iii).

It remains to argue that this algorithm can be implemented in NC^1. Recall that Step (ii) does not depend on the size of $\mathcal{A}$ and can thus be implemented in constant time (provided that Step (i) has been performed). Step (i) can be implemented in constant time as well, as the number of equivalence classes depends only on $\mathcal{T}$, and thus, constant: (polynomially many) processor units P_{abi}, for $(a,b,i) \in \mathsf{ind}(\mathcal{A})^2 \times \mathsf{tem}(\mathcal{A})$. For Step (iii), it suffices to prove that deciding whether a finite word is a possible prefix for $L(\mathfrak{A})$ (with a fixed Büchi automaton $\mathfrak{A}$) is in NC^1. As every regular language is in NC^1 [Straubing, 1994], it suffices to show that the set of all possible prefixes for $L(\mathfrak{A})$ is regular.

Let $\mathfrak{A} = (Q, \Sigma, q_0, \Delta, F)$ be a Büchi automaton, where all states are *terminating* in the sense that, for each $q \in Q$, there is a word $\sigma_0\sigma_1 \cdots \in L(\mathfrak{A})$ and an accepting run $q_0q_1 \cdots$ involving q (this assumption can be made without loss of generality). Construct an NFA $\mathfrak{B} = (Q, \Sigma, q_0, \Delta', Q)$ by taking

$$\Delta' = \{(q, \tau', q') \mid (q, \tau, q') \in \Delta, \tau' \subseteq \tau\}.$$

Claim 4. $L(\mathfrak{B})$ is the set of all possible prefixes for $L(\mathfrak{A})$.

This finishes the description of the algorithm and the analysis of its complexity. ❏

Temporal Acyclicity

Theorem 10 *TAQ answering over temporally acyclic* $\mathcal{TEL}^{\circ}$ (*with rigid concepts*) *is* PTIME*-complete in data and combined complexity.*

Proof. The lower bounds are inherited from $\mathcal{EL}$. To prove the upper bounds, we establish a small quasimodel property.

Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a temporal KB with a temporally acyclic TBox $\mathcal{T}$ and $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ its canonical quasimodel. By definition, there is a map $\ell_\circ \colon \mathsf{CN} \to \mathbb{N}$ satisfying conditions (i)–(iii) for temporally acyclic TBoxes. Without loss of generality, we assume that $\ell_\circ$ is minimal in the sense that for all maps $\ell'_\circ$ witnessing temporal acyclicity of $\mathcal{T}$, we have $\max_{A \in \mathsf{CN}} \ell_\circ(A) \leq \max_{A \in \mathsf{CN}} \ell'_\circ(A)$. It follows that $\min_{A \in \mathsf{CN}} \ell_\circ(A) = 0$. Let $n_\mathcal{T} = \max_{A \in \mathsf{CN}} \ell_\circ(A)$. By our assumption, $n_\mathcal{T} \leq |\mathcal{T}|$. Denote by l and u the numbers $\min \mathcal{A}$ and $\max \mathcal{A}$, respectively. We prove that the traces feature a *small quasimodel property* in the sense that, for every $d \in D$, there is some $\sigma_d \subseteq \mathsf{CN}$ such that

(P1) $\pi_a(j) = \sigma_a$ for all j with $j > u + n_\mathcal{T}$ or $j < l - n_\mathcal{T}$ and all $a \in \mathsf{ind}(\mathcal{A})$;

(P2) $\pi_B(j) = \sigma_B$ for all $j > n_\mathcal{T}$ or $j < -n_\mathcal{T}$ and $B \in \mathsf{CN}$.

Intuitively, σ_d is the set of concepts which contain d at all time instants, that is, the set of concepts which are 'rigid' for d. This property means that the canonical quasimodel has a restricted temporal extension that can change only $|\mathcal{T}|$ time points beyond the ABox.

We prove Properties **(P1)** and **(P2)** by giving an algorithm that constructs a small representation of the canonical quasimodel. The property will then follow from Lemma 17 below, which is part of the completeness proof of the algorithm.

The algorithm is an extension of the quasimodel construction procedure. Besides the finite parts $\hat{\pi}_d$ of the π_d, it maintains additional maps σ_d, which are initially empty, and for which we have the following completion rules for all $d \in D$ and all j with $l - n_\mathcal{T} \leq j \leq u + n_\mathcal{T}$:

1. if $A \in \hat{\pi}_d(j)$ and A is a *rigid* concept, then $A \in \sigma_d$;
2. $\sigma_d \subseteq \hat{\pi}_d(j)$;
3. if $A, A' \in \sigma_d$ and $A \sqcap A' \sqsubseteq B \in \mathcal{T}$, then $B \in \sigma_d$;
4. if $A \in \sigma_d$ and $\bigcirc_* A \sqsubseteq B \in \mathcal{T}$, then $B \in \sigma_d$;
5. if $r(a, b, n) \in \mathcal{A}$ and $r \in \mathsf{N}_\mathsf{R}^\mathsf{rig}$, then $\mathsf{proj}_r(\sigma_b) \subseteq \sigma_a$;[3]
6. if $A \in \sigma_d$, $A \sqsubseteq \exists r.B$, $r \in \mathsf{N}_\mathsf{R}^\mathsf{loc}$, then $\mathsf{proj}_r(\pi_B(0)) \subseteq \sigma_d$;
7. if $A \in \sigma_d$, $A \sqsubseteq \exists r.B$, $r \in \mathsf{N}_\mathsf{R}^\mathsf{rig}$, then $\mathsf{proj}_r(\pi_B(j)) \subseteq \sigma_d$;
8. if $A \in \pi_d(j)$, $A \sqsubseteq \exists r.B$, $r \in \mathsf{N}_\mathsf{R}^\mathsf{rig}$, then $\mathsf{proj}_r(\sigma_B) \subseteq \sigma_d$.

Note further that also rules **(t1)**, **(t2)**, **(q1)**–**(q4)** are applied to $\hat{\pi}_d$ in the range of $l - n_\mathcal{T} \leq j \leq u + n_\mathcal{T}$. The algorithm returns "yes" on query $A(a, i)$ iff $A \in \hat{\pi}_a(i)$.

It is routine to verify that the algorithm is sound. For completeness, assume that $A \notin \hat{\pi}_a(i)$. We define a collection of traces π_d, $d \in D$, by taking

$$\pi_d(i) = \begin{cases} \hat{\pi}_d(i), & \text{if } d \in \mathsf{ind}(\mathcal{A}) \text{ and } l - n_\mathcal{T} \leq i \leq u + n_\mathcal{T}, \\ \hat{\pi}_d(i) & \text{if } d \in \mathsf{CN} \text{ and } 0 \leq \ell_\bigcirc(d) + i \leq n_\mathcal{T}, \\ \sigma_d(i), & \text{otherwise,} \end{cases}$$

and verify that $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ is a quasimodel. This suffices since $A \notin \pi_a(i)$ and Theorem 4 yields $(\mathcal{T}, \mathcal{A}) \not\models A(a, i)$. Although Theorem 4 does not mention rigid concepts, they can be expressed in *full* $\mathcal{TEL}^\bigcirc$. It thus suffices to note that $\mathfrak{Q}$ respects rigid concepts: assume $A \in \pi_d(j)$ and A is a rigid concept name; if $A \in \sigma_d$, then $A \in \pi(i)$ for all $i \in \mathbb{Z}$ by rule 2; if $A \notin \sigma_d$, then $A \in \hat{\pi}_d(i)$ for some i. But then, by rule 1, $A \in \sigma_d$, a contradiction.

For showing that $\mathfrak{Q}$ is a quasimodel, we need to establish the following invariants of the maps $\hat{\pi}_d$ and σ_d.

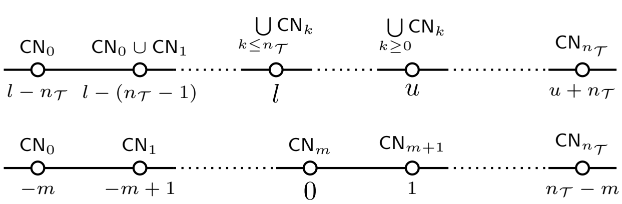
Lemma 17 *Let* $\mathsf{CN}_k = \{A \in \mathsf{CN} \mid \ell_\bigcirc(A) = k\}$. *Then, for every* $a \in \mathsf{ind}(\mathcal{A})$, *we have*

(a) $\hat{\pi}_a(j) \subseteq \bigcup_{k \geq j-u} \mathsf{CN}_k \cup \sigma_a$, *for all* j *with* $u \leq j \leq u + n_\mathcal{T}$;

(b) $\hat{\pi}_a(j) \subseteq \bigcup_{k \leq l-j} \mathsf{CN}_k \cup \sigma_a$, *for all* j *with* $l - n_\mathcal{T} \leq j \leq l$.

For every $B \in \mathsf{CN}$ *with* $\ell_\bigcirc(B) = m$, *we have*

(c) $\hat{\pi}_B(j) \subseteq \mathsf{CN}_{m+j} \cup \sigma_B$, *for all* j *with* $0 \leq m + j \leq n_\mathcal{T}$.



Proof. We prove these properties by induction on the number of rule applications in the construction of the canonical quasimodel. On empty maps, the three properties are trivially satisfied. Further note that rules 1–8 preserve the invariants since they only extend σ_d (except for rule 2, where the claim is also clear). Moreover, it is immediate that any application of **(q1)** or **(q2)** trivially preserves (a)–(c).

For **(q3)**, we need to verify only (a) and (b). Assume that $r(a, b, n) \in \mathcal{A}$ and some $B \in \mathsf{CN}$ is added to $\hat{\pi}_a(j)$ for some j with $l - n_\mathcal{T} \leq j \leq u + n_\mathcal{T}$, that is, there is a $B' \in \hat{\pi}_b(j)$ and $\exists r.B' \sqsubseteq B \in \mathcal{T}$. If $r \in \mathsf{N}_\mathsf{R}^\mathsf{loc}$ is a local role, then $j = n \in \mathsf{tem}(\mathcal{A})$, and so (a) and (b) are trivially satisfied. Otherwise, r is a rigid role. We show (a). By the induction hypothesis, either $B' \in \bigcup_{k \leq l-j} \mathsf{CN}_k$ or $B' \in \sigma_b$. In the former case, $B \in \bigcup_{k \leq l-j} \mathsf{CN}_k$, by the definition of $\ell_\bigcirc$. In the latter case, we obtain $B \in \sigma_a$ by rule 5; thus, $B \in \hat{\pi}_a(k)$ for all k with $l - n_\mathcal{T} \leq k \leq u + n_\mathcal{T}$. Hence, (a) is preserved; (b) is considered similarly.

For **(q4)**, assume that $A \in \hat{\pi}_d(n)$, $A \sqsubseteq \exists r.B \in \mathcal{T}$, $B' \in \hat{\pi}_B(i)$ and $\exists r.B' \sqsubseteq A' \in \mathcal{T}$. If r is a local role, then we need to consider $i = 0$ only, but then it suffices to note that $\ell_\bigcirc(A) = \ell_\bigcirc(B) = \ell_\bigcirc(B') = \ell_\bigcirc(A')$. If r is rigid, we distinguish cases:

– If $A \in \sigma_d$, then, by rule 7, we have $A' \in \sigma_d$;

– If $B' \in \sigma_B$, then, by rule 8, we have $A' \in \sigma_d$;

– If neither of the first two cases applies, $d \in \mathsf{ind}(\mathcal{A})$, and $j \geq u$ ($j < l$ and $l \leq j \leq u$ are similar), then, by the induction hypothesis, we have $\ell_\bigcirc(A) \geq j - u$ and $\ell_\bigcirc(B') = \ell_\bigcirc(B) + i = \ell_\bigcirc(A) + i$. Thus, $\ell_\bigcirc(A') = \ell_\bigcirc(B') \geq j + i - u$.

– If neither of the first two cases applies and $d \in \mathsf{CN}$, then, by the induction hypothesis, $\ell_\bigcirc(A) = \ell_\bigcirc(d) + j$ and $\ell_\bigcirc(B') = \ell_\bigcirc(B) + i = \ell_\bigcirc(A) + i$. Thus, we have $\ell_\bigcirc(A') = \ell_\bigcirc(B') = \ell_\bigcirc(d) + i + j$.

It remains to verify that **(t1)** and **(t2)** preserve (a)–(c). We show (a); the other properties are analogous. Let $B \in \hat{\pi}_a(j)$ and $\bigcirc_F B \sqsubseteq A \in \mathcal{T}$. By the induction hypothesis, either $B \in \bigcup_{k \geq j-u} \mathsf{CN}_k$ or $B \in \sigma_a$. In the former case, we have $\ell_\bigcirc(B) \geq j - u$, whence, by the definition of $\ell_\bigcirc$, we get $\ell_\bigcirc(A) = \ell_\bigcirc(B) - 1 \geq (j-1) - u$, and thus $A \in \bigcup_{k \geq (j-1)-u} \mathsf{CN}_k$. In the latter case, we obtain $A \in \sigma_a$ by rule 4. In either case, (a) is preserved. This finishes the proof of Lemma 17. ❑

We are now ready to show that $\mathfrak{Q}$ is a quasimodel. By construction, the π_d satisfy **(q1)**, **(q2)**, and **(t1)**. For **(t2)**, let $d \in \mathsf{ind}(\mathcal{A})$; the case $d \in \mathsf{CN}$ is analogous. Assume that $A \in \pi_d(j)$ and $\bigcirc_* A \sqsubseteq B \in \mathcal{T}$. If $l - n_\mathcal{T} < j < u + n_\mathcal{T}$, then **(t2)** is satisfied because $\hat{\pi}_d$ is closed under **(t2)** on this interval. If $j > u + n_\mathcal{T}$ or $j < l - n_\mathcal{T}$, then **(t2)** is due to rule 4. If $j = u + n_\mathcal{T}$, then, by Lemma 17, $\ell_\bigcirc(A) = n_\mathcal{T}$, whence $\ell_\bigcirc(B) < n_\mathcal{T}$, and so **(t2)** holds because $\hat{\pi}_d$ is closed under **(t2)**.

For **(q3)**, observe first that it is trivially satisfied in case r is a local role or $l - n_\mathcal{T} \leq j \leq u + n_\mathcal{T}$. So assume that r is rigid, take $j > u + n_\mathcal{T}$ (the case $j < l - n_\mathcal{T}$ is symmetric), $B \in \pi_b(j)$, and $\exists r.B \sqsubseteq B' \in \mathcal{T}$. By the definition of π_b, we have $B \in \sigma_b$. By rule 5, $B' \in \sigma_a$, and so $B' \in \pi_a(j)$, by the definition of π_a.

For **(q4)**, assume that $A \in \pi_d(j)$, $A \sqsubseteq \exists r.B \in \mathcal{T}$, $B' \in \pi_B(i)$ for some $i \in \mathbb{Z}$, and $\exists r.B' \sqsubseteq A' \in \mathcal{T}$. If r is a local role, then we need to consider $i = 0$ only. If $l - n_\mathcal{T} \leq j \leq u + n_\mathcal{T}$, then π_d is closed under **(q4)** becauae $\hat{\pi}_d$ is closed under **(q4)** on that interval. If $j > u + n_\mathcal{T}$

[3] proj applied to a set of concepts (rather than a trace) is defined in the obvious way.

($j < l - n_\mathcal{T}$), then we obtain $A \in \sigma_d$ and, by rule 6, $A' \in \sigma_d$, and hence $A' \in \pi_d(j)$. Finally, consider the following cases where r is rigid.

– If $A \in \sigma_d$, then $A' \in \sigma_d$ by rules 7 and 2.

– If $B' \in \sigma_B$, then, by rule 8, $A' \in \sigma_d$, and so, we obtain $A' \in \pi_d(j+i)$ by rule 2 and the definition of π_d;

– If neither of the two cases above applies, then, by Lemma 17, $0 \leq \ell_\bigcirc(B') \leq n_\mathcal{T}$ and $\ell_\bigcirc(B') = \ell_\bigcirc(B) + i$, and either $d \in \mathsf{Ind}(\mathcal{A})$ and $l - n_\mathcal{T} \leq j \leq u + n_\mathcal{T}$ or $d \in \mathsf{CN}$ and $0 \leq \ell_\bigcirc(A) + i \leq n_\mathcal{T}$. However, in both cases, **(q4)** is applied to $\hat{\pi}_d$ and $\hat{\pi}_B$, so we are done.

Hence, $\mathfrak{Q}$ is indeed a quasimodel. It remains to note that the algorithm terminates after polynomially many steps: the bounds on the size of the $\hat{\pi}_d$ and σ_d imply that they can be represented in polynomial space, and the completion rules extend these data structures at each step. ❑

Inflationary $\mathcal{TEL}^\diamond$

Theorem 11 *TAQ answering over $\mathcal{TEL}^\diamond_{\mathsf{infl}}$ is* PTIME*-complete in both data and combined complexity.*

Proof. The lower bounds are inherited from $\mathcal{EL}$. To prove the upper bounds, we establish some key properties of traces in the canonical quasimodel over $\mathcal{TEL}^\diamond_{\mathsf{infl}}$-TBoxes.

Let $(\mathcal{T}, \mathcal{A})$ be a KB with a $\mathcal{TEL}^\diamond_{\mathsf{infl}}$-TBox $\mathcal{T}$ and let $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ be its canonical quasimodel. We remind the reader that, by definition, traces for $\mathcal{T}$ are closed under **(t1)** for $\sqcap$ and **(t2′)** for $\diamond_P$ and $\diamond_F$. We are going to show in Lemma 18 below that the traces π_d in the canonical quasimodel are in fact all bow ties.

Lemma 18 *If $\mathcal{T}$ is a $\mathcal{TEL}^\diamond_{\mathsf{infl}}$-TBox, then traces in the canonical quasimodel $\mathfrak{Q} = \{\pi_d \mid d \in D\}$ enjoy the following properties*:

(a) *π_a is a $[\min \mathcal{A}, \max \mathcal{A}]$-bow tie, for each $a \in \mathsf{ind}(\mathcal{A})$;*

(b) *π_B is a $[0,0]$-bow tie, for each $B \in \mathsf{CN}$.*

Proof. We show by induction on the number of rule applications that the traces in the quasimodel construction procedure satisfy (a) and (b). The basis of induction is immediate from **(q1)** and **(q2)**. For the inductive step, we make a case distinction on which rule is applied. For **(t1)** and **(t2′)**, we assume that they are always applied exhaustively. Since $\diamond_*$ are the only temporal operators allowed in $\mathcal{T}$, it can be readily verified, that

Claim. For any $[l,u]$-bow tie ϱ, the closure of ϱ under **(t1)** and **(t2′)** is an $[l,u]$-bow tie as well.

For **(q3)**, let $r(a,b,n) \in \mathcal{A}$. Property (b) is trivial and (a) is preserved because $\mathsf{proj}_r(\pi_b)$ is an $[l,u]$-bow tie for any $[l,u]$-bow tie π_b and the component-wise union of two $[l,u]$-bow ties is also an $[l,u]$-bow tie.

For **(q4)**, let $A \in \pi_d(n)$ and $A \sqsubseteq \exists r.B \in \mathcal{T}$. By the induction hypothesis, π_d is an $[l,u]$-bow tie for some l,u. Also observe that, by the induction hypothesis, π_B is a $[0,0]$-bow tie and therefore, for any $m \in \mathbb{Z}$, the map $\varrho_m : i \to \mathsf{proj}_r(\pi_B)(i-m)$ is an $[m,m]$-bow tie. We consider the following three cases.

1. If $l \leq n \leq u$, then the application of **(q4)** extends π_d with ϱ_n, which results in an $[l,u]$-bow tie, as required for (a) and (b).
2. If $n > u$, let $m_0 > u$ be minimal with $A \in \pi_d(m_0)$. Observe that **(q4)** is applicable to all $m \geq m_0$ and *all* such applications of **(q4)** can be captured by taking the union of π_d and *all* ϱ_m, for $m \geq m_0$. Observe that $\bigcup_{m \geq m_0} \varrho_m$ is in fact $\mathsf{proj}_r(\pi)$ for π defined by taking
$$\pi(n) = \begin{cases} X, & \text{if } n < m_0,\\ X \cup \pi_B(0) \cup \pi_B(n - m_0) & \text{if } n \geq m_0, \end{cases}$$
where X be the maximum set at 'the left end' of π_B: $X = \bigcup_{n<0} \pi_B(n)$. Since π_B is a $[0,0]$-bow tie, the map π is monotone, that is, $\pi(n) \subseteq \pi(n+1)$ for all $n \in \mathbb{Z}$, and $\pi(n) = \pi(n-1)$ for all $n < m_0$. It follows that also $\mathsf{proj}_r(\pi)$ is monotone and $\mathsf{proj}_r(\pi)(n) = \mathsf{proj}(\pi)(n-1)$ for all $n < m_0$. It is then easy to see that the component-wise union of π_d and $\mathsf{proj}_r(\pi)$ is an $[l,u]$-bow tie.
3. If $n < l$, then the construction is the mirror image of 2.

This completes the proof of Lemma 18. ❑

We are now in a position to complete the proof of Theorem 11. First, observe that the traces maintained by the procedure for constructing the canonical quasimodel of a KBs with a $\mathcal{TEL}^\diamond_{\mathsf{infl}}$-TBox can be represented using a polynomial amount of space because, by Lemma 18, they are $[l,u]$-bow ties with both $l,u \in \mathsf{tem}(\mathcal{A})$: one needs to store only the middle section and at most $|\mathsf{CN}|$ steps at both ends. Second, the traces are extended with every rule application, and so, the procedure must terminate after polynomially many steps, which gives PTIME upper complexity bound for both data and combined complexity. ❑

References

[Straubing, 1994] Straubing, H. 1994. *Finite Automata, Formal Logic, and Circuit Complexity*. Birkhauser Verlag.

[Vardi, 2006] Vardi, M. Y. 2006. Automata-theoretic techniques for temporal reasoning. In *Handbook of Modal Logic*, 971–989. Elsevier.