Temporal Description Logic for Ontology-Based Data Access (Extended Version)

Alessandro Artale\textsuperscript{1}, Roman Kontchakov\textsuperscript{2}, Frank Wolter\textsuperscript{3} and Michael Zakharyaschev\textsuperscript{2}

\textsuperscript{1}Faculty of Computer Science
Free University of Bozen-Bolzano, Italy
\textsuperscript{2}Department of Computer Science and Information Systems
Birkbeck, University of London, U.K.
\textsuperscript{3}Department of Computer Science,
University of Liverpool, U.K.

May 1, 2013

Abstract

Our aim is to investigate ontology-based data access over temporal data with validity time and ontologies capable of temporal conceptual modelling. To this end, we design a temporal description logic, TQL, that extends the standard ontology language OWL 2 QL, provides basic means for temporal conceptual modelling and ensures first-order rewritability of conjunctive queries for suitably defined data instances with validity time.

1 Introduction

One of the most promising and exciting applications of description logics (DLs) is to supply ontology languages and query answering technologies for ontology-based data access (OBDA), a way of querying incomplete data sources that uses ontologies to provide additional conceptual information about the domains of interest and enrich the query vocabulary. The current W3C standard language for OBDA is OWL 2 QL, which was built on the DL-Lite family of DLs [Calvanese et al., 2006; Calvanese et al., 2007]. To answer a conjunctive query \( q \) over an OWL 2 QL ontology \( T \) and instance data \( A \), an OBDA system first ‘rewrites’ \( q \) and \( T \) into a new first-order query \( q' \) and then evaluates \( q' \) over \( A \) (without using the ontology). The evaluation task is performed by a conventional relational database management system. Finding efficient and practical rewritings has been the subject of extensive research [Pérez-Urbina et al., 2009; Rosati and Almatelli, 2010; Kontchakov et al., 2010; Chortaras et al., 2011; Gottlob et al., 2011; König et al., 2012].

Another fundamental feature of OWL 2 QL, supplementing its first-order rewritability, is the ability to capture basic conceptual data modelling constructs [Berardi et al., 2005; Artale et al., 2007].

In applications, instance data is often time-dependent: employment contracts come to an end, parliaments are elected, children are born. Temporal data can be modelled by pairs consisting of facts and their validity time; for example, \( \text{givesBirth}(\text{diana}, \text{william}, 1982) \). To query data with validity time, it would be useful to employ an ontology that provides a conceptual model for both static and temporal aspects of the domain of interest. Thus, when querying the fact above, one could use the knowledge that, if \( x \) gives birth to \( y \), then \( x \) becomes a mother of \( y \) from that moment on:

\[
\Diamond_{\tau} \text{givesBirth} \sqsubseteq \text{motherOf}, \tag{1}
\]

where \( \Diamond_{\tau} \) reads ‘sometime in the past’ OWL 2 QL does not support temporal conceptual modelling and, rather surprisingly, no attempt has yet been made to lift the OBDA framework to temporal ontologies and data.

Temporal extensions of DLs have been investigated since 1993; see [Gabbay et al., 2003; Lutz et al., 2008; Artale and Franconi, 2005] for surveys and [Franconi and Toman, 2011; Gutiérrez-Basulto and Klarman, 2012; Baader et al., 2012] for more recent developments. Temporalised DL-Lite logics have been constructed for
temporal conceptual data modelling [Artale et al., 2010]. But unfortunately, none of the existing temporal DLs supports first-order rewritability.

The aim of this paper is to design a temporal DL that contains OWL 2 QL, provides basic means for temporal conceptual modelling and, at the same time, ensures first-order rewritability of conjunctive queries (for suitably defined data instances with validity time).

The temporal extension TQL of OWL 2 QL we present here is interpreted over sequences $I(n), n \in \mathbb{Z}$, of standard DL structures reflecting possible evolutions of data. TBox axioms are interpreted globally, that is, are assumed to hold in all of the $I(n)$, but the concepts and roles they contain can vary in time. ABox assertions (temporal data) are time-stamped unary (for concepts) and binary (for roles) predicates that hold at the specified moments of time. Concept (role) inclusions of TQL generalise OWL 2 QL inclusions by allowing intersections of basic concepts (roles) in the left-hand side, possibly prefixed with temporal operators $\Diamond_i$ (sometime in the past) or $\Diamond_{\tau}$ (sometime in the future). Among other things, one can express in TQL that a concept/role name is rigid (or time-independent), persistent in the past/future or instantaneous. For example, $\Diamond_i \Diamond_{\tau} \text{Person} \sqsubseteq \text{Person}$ states that the concept Person is rigid, $\Diamond_{\tau} \text{hasName} \sqsubseteq \text{hasName}$ says that the role hasName is persistent in the future, while $\text{givesBirth} \sqcap \Diamond_{\tau} \text{givesBirth} \sqsubseteq \bot$ implies that givesBirth is instantaneous. Inclusions such as $\Diamond_{\tau} \text{Start} \sqcap \Diamond_{\tau} \text{End} \sqsubseteq \text{Employed}$ represent convexity (or existential rigidity) of concepts or roles. However, in contrast to most existing temporal DLs, we cannot use temporal operators in the right-hand side of inclusions (e.g., to say that every student will eventually graduate: $\text{Student} \sqsubseteq \Diamond_{\tau} \text{Graduate}$).

In conjunctive queries (CQs) over TQL knowledge bases, we allow time-stamped predicates together with atoms of the form $(\tau < \tau')$ or $(\tau = \tau')$, where $\tau, \tau'$ are temporal constants denoting integers or variables ranging over integers.

Our main result is that, given a TQL TBox $\mathcal{T}$ and a CQ $q$, one can construct a union $q'$ of CQs such that the answers to $q$ over $\mathcal{T}$ and any temporal ABox $\mathcal{A}$ can be computed by evaluating $q'$ over $\mathcal{A}$ extended with the temporal precedence relation $<$ between the moments of time in $\mathcal{A}$. For example, the query $\text{motherOf}(x, y, t)$ over (1) can be rewritten as

$$\text{motherOf}(x, y, t) \lor \exists t' ((t' < t) \land \text{givesBirth}(x, y, t')).$$

Note that the addition of the transitive relation $<$ to the ABox is unavoidable: without it, there exists no first-order rewriting even for the simple example above [Libkin, 2003 Cor. 4.13].

From a technical viewpoint, one of the challenges we are facing is that, in contrast to known OBDA languages with CQ rewritability (including fragments of datalog$^*$ [Calì et al., 2012]), witnesses for existential quantifiers outside the ABox are not independent from each other but interact via the temporal precedence relation. For this reason, a reduction to known languages appears to be impossible and a novel approach to rewriting has to be found. We also observe that straightforward temporal extensions of TQL lose first-order rewritability. For example, query answering over the ontology $\{\text{Student} \sqsubseteq \Diamond_{\tau} \text{Graduate}\}$ is shown to be non-tractable.

## 2 TQL: a Temporal Extension of OWL 2 QL

Concepts $C$ and roles $S$ of TQL are defined by the grammar:

$$R ::= \bot \mid P_i \mid P_i^\tau,$$
$$B ::= \bot \mid A_i \mid \exists R,$$
$$C ::= B \mid C \sqcap C \mid \Diamond_i C \mid \Diamond_{\tau} C,$$
$$S ::= R \mid S \sqcap S \mid \Diamond_i S \mid \Diamond_{\tau} S,$$

where $A_i$ is a concept name, $P_i$ a role name ($i \geq 0$), and $\Diamond_i$ and $\Diamond_{\tau}$ are temporal operators ‘sometime in the past’ and ‘sometime in the future,’ respectively. We call concepts and roles of the form $B$ and $R$ basic. A TQL TBox, $\mathcal{T}$, is a finite set of concept and role inclusions of the form

$$C \sqsubseteq B, \quad S \sqsubseteq R,$$
which are assumed to hold globally (over the whole timeline). Note that the $\diamond_{\text{r}, \text{m}}$-free fragment of TQL is an extension of the description logic DL-Lite$_h$ [Artale et al., 2009] with role inclusions of the form $R_1 \cap \cdots \cap R_n \sqsubseteq R$; it properly contains OWL 2 QL (the missing role constraints can be safely added to the language).

A TQL ABox, $A$, is a (finite) set of atoms $P_1(a, b, n)$ and $A_i(a, n)$, where $a, b$ are individual constants and $n \in \mathbb{Z}$ a temporal constant. The set of individual constants in $A$ is denoted by $\text{ind}(A)$, and the set of temporal constants by $\text{tem}(A)$. A TQL knowledge base (KB) is a pair $K = (T, A)$, where $T$ is a TBox and $A$ an ABox.

A temporal interpretation, $I$, is given by the ordered set $(\mathbb{Z}, <)$ of time points and standard (atemporal) interpretations $I(n) = (\Delta^I, \mathcal{I}(n))$, for each $n \in \mathbb{Z}$. Thus, $\Delta^I \neq \emptyset$ is the common domain of all $I(n)$, $a_i^{I(n)} \in \Delta^I$, $A_i^{I(n)} \subseteq \Delta^I$ and $P_i^{I(n)} \subseteq \Delta^I \times \Delta^I$. We assume that $a_i^{I(n)} = a_i^{I(0)}$, for all $n \in \mathbb{Z}$. To simplify presentation, we adopt the unique name assumption, that is, $a_i^{I(n)} \neq a_j^{I(n)}$ for $i \neq j$ (although the obtained results hold without it). The role and concept constructs are interpreted in $I$ as follows, where $n \in \mathbb{Z}$:

$$
\bot^{I(n)} = \emptyset \text{ (for both concepts and roles),}
$$

$$
\left\{P_i^{-}\right\}^{I(n)} = \left\{ (x, y) \mid (y, x) \in P_i^{I(n)} \right\},
$$

$$
\left\{\exists R\right\}^{I(n)} = \left\{ x \mid (x, y) \in R^{I(n)}, \text{ for some } y \right\},
$$

$$
(C_1 \cap C_2)^{I(n)} = C_1^{I(n)} \cap C_2^{I(n)},
$$

$$
\left\{\forall \sigma C\right\}^{I(n)} = \left\{ x \mid x \in C^{I(m)}, \text{ for some } m < n \right\},
$$

$$
\left\{\forall \sigma C\right\}^{I(n)} = \left\{ x \mid x \in C^{I(m)}, \text{ for some } m > n \right\},
$$

$$
(S_1 \cap S_2)^{I(n)} = S_1^{I(n)} \cap S_2^{I(n)},
$$

$$
\left\{\forall \sigma S\right\}^{I(n)} = \left\{ (x, y) \mid (x, y) \in S^{I(m)}, \text{ for some } m < n \right\},
$$

$$
\left\{\forall \sigma S\right\}^{I(n)} = \left\{ (x, y) \mid (x, y) \in S^{I(m)}, \text{ for some } m > n \right\}.
$$

The satisfaction relation $|=\!=$ is defined by taking

$$
I \models A_i(a, n) \quad \text{iff} \quad a^{I(n)} \in A_i^{I(n)},
$$

$$
I \models P_i(a, b, n) \quad \text{iff} \quad (a^{I(n)}, b^{I(n)}) \in P_i^{I(n)},
$$

$$
I \models C \subseteq B \quad \text{iff} \quad C^{I(n)} \subseteq B^{I(n)}, \text{ for all } n \in \mathbb{Z},
$$

$$
I \models S \subseteq R \quad \text{iff} \quad S^{I(n)} \subseteq R^{I(n)}, \text{ for all } n \in \mathbb{Z}.
$$

If all inclusions in $T$ and atoms in $A$ are satisfied in $I$, we call $I$ a model of $K = (T, A)$ and write $I \models K$.

A conjunctive query (CQ) is a (two-sorted) first-order formula $q(\vec{x}, \vec{s}) = \exists \vec{y}, \vec{z} \varphi(\vec{x}, \vec{y}, \vec{s}, \vec{z})$, where $\varphi(\vec{x}, \vec{y}, \vec{s}, \vec{z})$ is a conjunction of atoms of the form $A_i(\xi, \tau)$, $P_i(\xi, \zeta, \tau)$, $(\tau = \sigma)$ and $(\tau < \sigma)$, with $\xi$, $\zeta$ being individual terms—individual constants or variables in $\vec{x}, \vec{y}$—and $\tau, \sigma$ temporal terms—temporal constants or variables in $\vec{t}, \vec{s}$. In a positive existential query (PEQ) $q$, the formula $\varphi$ can also contain $\forall$. A union of CQs (UCQ) is a disjunction of CQs (so every PEQ is equivalent to an exponentially larger UCQ).

Given a KB $K = (T, A)$ and a CQ $q(\vec{x}, \vec{s})$, we call tuples $\vec{a} \subseteq \text{ind}(A)$ and $\vec{n} \subseteq \text{tem}(A)$ a certain answer to $q(\vec{x}, \vec{s})$ over $K$ and write $K \models q(\vec{a}, \vec{n})$, if $I \models q(\vec{a}, \vec{n})$ for every model $I$ of $K$ (understood as a two-sorted first-order model).

**Example 1** Suppose Bob was a lecturer at UCL between times $n_1$ and $n_2$, after which he was appointed professor on a permanent contract. To model this situation, we use individual names, $e_1$ and $e_2$, to represent the two events of Bob’s employment. The ABox will contain $n_1 < n_2$ and the atoms $\text{lect}(\text{bob}, e_1, n_1)$, $\text{lect}(\text{bob}, e_1, n_2)$, $\text{prof}(\text{bob}, e_2, n_2 + 1)$. In the TBox, we make sure that everybody is holding the corre-
spending post over the duration of the contract, and include other knowledge about the university life:

\[ \diamond_q \text{lect} \sqcap \diamond_q \text{lect} \subseteq \text{lect}, \]
\[ \exists q \text{lect} \sqsubseteq \text{Lecturer}, \]
\[ \text{Professor} \sqsubseteq \exists q \text{supervisesPhD}, \]
\[ \diamond_q \text{supervisesPhD} \sqcap \diamond_q \text{supervisesPhD} \subseteq \text{supervisesPhD}, \]

etc.

We can now obtain staff who supervised PhDs between times \( k_1 \) and \( k_2 \) by posing the following CQ:

\[ \exists y, t \left( (k_1 < t < k_2) \land \text{Staff}(x, t) \land \text{supervisesPhD}(x, y, t) \right). \]

The key idea of OBDA is to reduce answering CQs over KBs to evaluating FO-queries over relational databases. To obtain such a reduction for TQL KBs, we employ a very basic type of temporal databases. With every TQL ABox \( \mathcal{A} \), we associate a data instance \( \mathcal{A} \) which contains all atoms from \( \mathcal{A} \) as well as the atoms \( (n_1 < n_2) \) such that \( n_1 \in \mathbb{Z} \) with \( \min \text{tem}(\mathcal{A}) \leq n_1 \leq \max \text{tem}(\mathcal{A}) \) and \( n_1 < n_2 \). Thus, in addition to \( \mathcal{A} \), we explicitly include in \( \mathcal{A} \) the temporal precedence relation over the convex closure of the time points that occur in \( \mathcal{A} \). (Note that, in standard temporal databases, the order over timestamps is built-in.)

The main result of this paper is the following:

**Theorem 2** Let \( q(\vec{x}, \vec{s}) \) be a CQ and \( T \) a TQL TBox. Then one can construct a UCQ \( q'(\vec{x}, \vec{s}) \) such that, for any consistent KB \( (T, \mathcal{A}) \) such that \( \mathcal{A} \) contains all temporal constants from \( q \), any \( \vec{a} \subseteq \text{ind}(\mathcal{A}) \) and \( \vec{n} \subseteq \text{tem}(\mathcal{A}) \), we have \( (T, \mathcal{A}) \models q(\vec{a}, \vec{n}) \iff [\mathcal{A}] \models q'(\vec{a}, \vec{n}). \)

Such a UCQ \( q'(\vec{x}, \vec{s}) \) is called a rewriting for \( q \) and \( T \). We begin by showing how to compute rewritings for CQs over KBs with empty TBoxes.

For an ABox \( \mathcal{A} \), we denote by \( \mathcal{A}^\infty \) the infinite data instance which contains the atoms in \( \mathcal{A} \) as well as all \( (n_1 < n_2) \) such that \( n_1, n_2 \in \mathbb{Z} \) and \( n_1 < n_2 \). It will be convenient to regard CQs \( q(\vec{x}, \vec{s}) \) as sets of atoms, so that we can write, e.g., \( A(\xi, \tau) \in q \). We say that \( q \) is totally ordered if, for any temporal terms \( \tau, \tau' \) in \( q \), at least one of the constraints \( \tau < \tau' \) or \( \tau' < \tau \) is in \( q \) and the set of such constraints is consistent (in the sense that it can be satisfied in \( \mathbb{Z} \)). Clearly, every CQ is equivalent to a union of totally ordered CQs (note that the empty union is \( \bot \)).

**Lemma 3** For every UCQ \( q(\vec{x}, \vec{s}) \), one can compute a UCQ \( q'(\vec{x}, \vec{s}) \) such that, for any ABox \( \mathcal{A} \) containing all temporal constants from \( q \) and any \( \vec{a} \subseteq \text{ind}(\mathcal{A}), \vec{n} \subseteq \text{tem}(\mathcal{A}) \), we have

\[ \mathcal{A}^\infty \models q(\vec{a}, \vec{n}) \iff [\mathcal{A}] \models q'(\vec{a}, \vec{n}). \]

**Proof.** We assume that every CQ \( q_0 \) in \( q \) is totally ordered. In each such \( q_0 \), we remove a bound temporal variable \( t \) together with the atoms containing \( t \) if at least one of the following two conditions holds:

– there is no temporal constant or free temporal variable \( \tau \) with \( (\tau < t) \in q_0 \), and for no temporal term \( \tau' \) and atom of the form \( A(\xi, \tau') \) or \( P(\xi, \zeta, \tau') \) in \( q_0 \) do we have \( (\tau' < t) \) or \( (\tau' = t) \) in \( q_0 \);

– the same as above but with \( < \) replaced by \( > \).

It is readily checked that the resulting UCQ is as required.

**Example 4** Suppose \( T = \{ \diamond q C \sqsubseteq A, \diamond q A \sqsubseteq B \} \) and \( q(x, s) = B(x, s) \). Then, for any \( A, a \in \text{ind}(A), n \in \text{tem}(A) \), we have \( (T, \mathcal{A}) \models q(a, n) \iff [\mathcal{A}] \models q'(a, n) \), where

\[ q'(x, s) = B(x, s) \lor \exists t ((t < s) \land A(x, t)) \lor \exists t, r ((t < s) \land (t < r) \land C(x, r)). \]

Note, however, that \( q' \) is not a rewriting for \( q \) and \( T \). Take, for example, \( \mathcal{A} = \{ C(a, 0) \} \). Then \( (T, \mathcal{A}) \models B(a, 0) \) but \( [\mathcal{A}] \not\models q'(a, 0) \). A correct rewriting is obtained by replacing the last disjunct in \( q' \) with \( \forall r C(x, r) \); it can be computed by applying Lemma 3 to \( q' \) and slightly simplifying the result.
Lemma 7 Suppose \( \text{cl} \) holds.
The problem of finding rewritings for CQs and TQL TBoxes can be reduced to the case where the TBoxes only contain inclusions of the form

\[
B_1 \cap B_2 \subseteq B, \quad \Leftarrow B_1 \subseteq B_2, \quad \Leftarrow B_1 \subseteq B_2,
\]

\[
R_1 \cap R_2 \subseteq R, \quad \Leftarrow R_1 \subseteq R_2, \quad \Leftarrow R_1 \subseteq R_2.
\]

We say that such TBoxes are in normal form.

Theorem 5 For every TQL TBox \( T \), one can construct in polynomial time a TQL TBox \( T' \) in normal form (possibly containing additional concept and role names) such that \( T' \models T \) and, for every model \( \mathcal{I} \) of \( T \), there exists a model of \( T' \) that coincides with \( \mathcal{I} \) on all concept and role names in \( T \).

Suppose now that we have a UCQ rewriting \( q' \) for a CQ \( q \) and the TBox \( T' \) in Theorem 5. We obtain a rewriting for \( q \) and \( T \) simply by removing from \( q' \) those CQs that contain symbols occurring in \( T' \) but not in \( T \). From now on, we assume that all TQL TBoxes are in normal form. The set of role names in \( T \) and with their inverses is denoted by \( R_T \), while \( |T| \) is the number of concept and role names in \( T \).

We begin the construction of rewritings by considering the case when all concept inclusions are of the form \( C \subseteq A \), so existential quantification \( \exists R \) does not occur in the right-hand side. TQL TBoxes of this form will be called flat. Note that RDFS statements can be expressed by means of flat TBoxes.

3 UCQ Rewriting for Flat TBoxes

Let \( \mathcal{K} = (T, A) \) be a KB with a flat TBox \( T \) (in normal form). Our first aim is to construct a model \( \mathcal{C}_\mathcal{K} \) of \( \mathcal{K} \), called the canonical model, for the following theorem holds:

Theorem 6 For any consistent KB \( \mathcal{K} = (T, A) \) with flat \( T \) and any CQ \( q(x, z) \), we have \( \mathcal{K} \models q(\bar{a}, \bar{n}) \) iff \( \mathcal{C}_\mathcal{K} \models q(\bar{a}, \bar{n}) \), for all tuples \( \bar{a} \in \text{ind}(A) \) and \( \bar{n} \in \mathbb{Z} \).

The construction uses a closure operator, \( \text{cl} \), which applies the rules (ex), (c1)-(c3), (r1)-(r3) below to a set, \( S \), of atoms of the form \( R(u, v, n), A(u, n), \exists R(u, n) \) or \( (n < n') \); \( \text{cl}(S) \) is the result of (non-recursively) applying those rules to \( S \),

\[
\text{cl}^0(S) = S, \quad \text{cl}^{i+1}(S) = \text{cl}(\text{cl}^i(S)), \quad \text{cl}^\infty(S) = \bigcup_{i \geq 0} \text{cl}^i(S).
\]

(ex) If \( R(u, v, n) \in S \) then add \( \exists R(u, n), \exists R^-(v, n) \) to \( S \);

(c1) if \( (B_1 \cap B_2 \subseteq B) \in T \) and \( B_1(u, n), B_2(u, n) \in S \), then add \( B(u, n) \) to \( S \);

(c2) if \( (\Leftarrow B \subseteq B') \in T \), \( B(u, m) \in S \) for some \( m < n \) and \( n \) occurs in \( S \), then add \( B'(u, n) \) to \( S \);

(c3) if \( (\Leftarrow B \subseteq B') \in T \), \( B(u, m) \in S \) for some \( m > n \) and \( n \) occurs in \( S \), then add \( B'(u, n) \) to \( S \);

(r1) if \( (R_1 \cap R_2 \subseteq R) \in T \) and \( R_1(u, v, n), R_2(u, v, n) \) are in \( S \), then add \( R(u, v, n) \) to \( S \);

(r2) if \( (\Leftarrow R \subseteq R') \in T \), \( R(u, v, m) \in S \) for some \( m < n \) and \( n \) occurs in \( S \), then add \( R'(u, v, n) \) to \( S \);

(r3) if \( (\Leftarrow R \subseteq R') \in T \), \( R(u, v, m) \in S \) for some \( m > n \) and \( n \) occurs in \( S \), then add \( R'(u, v, n) \) to \( S \).

Note first that \( \mathcal{K} = (T, A) \) is inconsistent iff \( \bot \in \text{cl}^\infty(\mathcal{A}) \). If \( \mathcal{K} \) is consistent, we define the canonical model \( \mathcal{C}_\mathcal{K} \) of \( \mathcal{K} \) by taking \( \Delta_{\mathcal{C}_\mathcal{K}} = \text{ind}(A) \), \( a \in A_{\mathcal{C}_\mathcal{K}}^{\infty} \) iff \( \mathcal{C}_\mathcal{K} \models A(a, n) \in \text{cl}^\infty(\mathcal{A}) \), and \( (a, b) \in P_{\mathcal{C}_\mathcal{K}}^{\infty} \) iff \( \mathcal{C}_\mathcal{K} \models P(a, b, n) \in \text{cl}^\infty(\mathcal{A}) \), for \( n \in \mathbb{Z} \). (As \( T \) is flat, atoms of the form \( \exists R(u, n) \) can only be added by (ex).) This gives us Theorem 6. The following lemma shows that to construct \( \mathcal{C}_\mathcal{K} \) we actually need only a bounded number of applications of \( \text{cl} \) which does not depend on \( A \):

Lemma 7 Suppose \( T \) is a flat TBox, let \( n_T = (4 \cdot |T|)^4 \). Then \( \text{cl}^n_T(\mathcal{A}) = \text{cl}_{n_T}(\mathcal{A}) \), for any ABox \( A \).
 Proof. It is not hard to see that cl^{∞}(S) can be obtained by first exhaustively applying (r1)–(r3), then (ex), and after that (c1)–(c3). Since no recursion of (ex) is needed, it is sufficient to bound the recursion depth for applications of (r1)–(r3) and (c1)–(c3) separately. As both behave similarly, we focus on (r1)–(r3).

We now use Lemma 7 to construct a rewriting for any flat TBox T and CQ q(⃗x, ⃗s). For a concept C and a role S, denote by C^↓ and S^↓ their standard FO-translations: e.g., (\langle Q A \rangle^↑) = \exists t ((\tau < t) \land A(\xi, t)) and (\langle R \rangle^↑(\xi, \tau) = \exists y R(\xi, y, \tau). Now, given a PEQ \varphi, we set \varphi^↓ = \varphi and define, inductively, \varphi^{(n+1)}↓ as the result of replacing every

- A(\xi, \tau) \land A(\xi, \tau) \lor V_{(C\subseteq A)\in T}(C^↓(\xi, \tau))^n↓,
- P(\xi, \zeta, \tau) \lor V_{(S\subseteq P)\in T}(S^↓(\xi, \zeta, \tau))^n↓.

Finally, we set

\text{ext}^\varphi_q(\vec{x}, \vec{s}) = (q(\vec{x}, \vec{s}))^{\varphi↓↓}.

Clearly, \text{ext}^\varphi_q(\vec{x}, \vec{s}) is a PEQ, and so can be equivalently transformed into a UCQ. Denote by T^↓ the result of replacing \Lambda with a fresh concept name, say F, in all concept inclusions and with a fresh role name, say Q, in all role inclusions of T. Clearly (T^↓, \Lambda) is consistent for any ABox \Lambda. Let q^↓ = (\exists x, t F(x, t)) \lor (\exists x, y, t Q(x, y, t)). By Theorem 6 and Lemma 7 we obtain:

**Theorem 8** Let T be a flat TBox and q(⃗x, ⃗s) a CQ. Then, for any consistent KB (T, A), any \vec{a} \subseteq \text{ind}(A) and \vec{n} \subseteq \mathbb{Z},

(T, A) \models q(\vec{a}, \vec{n}) \iff \Lambda^\varphi \models \text{ext}^\varphi_q(\vec{a}, \vec{n}).

(T, A) is inconsistent iff (T^↓, \Lambda) \models q^↓.

Thus, we obtain a rewriting for q and T using Lemma 3.

4 Canonical Models for Arbitrary TBoxes

Canonical models for consistent KBs K = (T, A) with not necessarily flat TBoxes T (in normal form) can be constructed starting from A^↓ and using the rules given in the previous section together with the following one:

(\therefore) if \exists R(u, v) \in S and R(u, v, n) \notin S for any v, then add R(u, v, n) to S, for some fresh individual name v; in this case we write u \sim_n^R v.

Denote by cl the closure operator under the resulting 8 rules. Again, K is consistent iff \perp \in cl^{∞}(A^↓). If K is consistent, we define the canonical model \mathcal{C}_K for K by setting \text{cl}^{∞}(A^↓) in the same way as in Section 5 but taking the domain \Delta^{\mathcal{C}_K} to contain all the individual names in \text{cl}^{∞}(A^↓).

**Theorem 9** For every consistent K = (T, A) and every CQ q(⃗x, ⃗s), we have K \models q(\vec{a}, \vec{n}) iff \mathcal{C}_K \models q(\vec{a}, \vec{n}), for any tuples \vec{a} \subseteq \text{ind}(A) and \vec{n} \subseteq \mathbb{Z}.

**Example 10** Let K = (T, A) with A = \{A(\alpha, 0)\} and

\[ T = \{ A \subseteq R, \exists \alpha R \subseteq Q, \exists Q' \subseteq \exists S, \exists Q' \subseteq P, \exists \alpha S \subseteq S' \}. \]

A fragment of the model \mathcal{C}_K is shown in the picture below:
We say that the individuals \( a \in \text{ind}(A) \) are of depth 0 in \( C_K \); now, if \( u \) is of depth \( d \) in \( C_K \) and \( a \leadsto_R^n u \), for some \( n \in \mathbb{Z} \) and \( R \), then \( v \) is of depth \( d + 1 \) in \( C_K \). Thus, both \( u_1 \) and \( u_2 \) in Example 10 are of depth 2 and \( v \) is of depth 1. The restriction of \( C_K \), treated as a set of atoms, to the individual names of depth \( \leq d \) is denoted by \( C_K^d \). Note that this set is not necessarily closed under the rule \((\leadsto)\).

In the remainder of this section, we describe the structure of \( C_K \), which is required for the rewriting in the next section. We split \( C_K \) into two parts: one consists of the elements in \( \text{ind}(A) \), while the other contains the fresh individuals introduced by \((\leadsto)\). As this rule always uses fresh individuals, to understand the structure of the latter part it is enough to consider KBs of the form \( K_P^f = (\mathcal{T} \cup \{A \subseteq \exists R\}, \{A(a,0)\}) \) with fresh \( A \). We begin by analysing the behaviour of the atoms \( R'(a, u, n) \) entailed by \( R(a, u, 0) \), where \( a \leadsto_R^n u \).

**Lemma 11 (monotonicity)** Suppose \( a \leadsto_R^0 u \) in \( C_{K,T,R} \). If either \( m < n < 0 \) or \( 0 < n < m \), then \( R'(a, u, n) \in C_{K,T,R} \) implies \( R'(a, u, m) \in C_{K,T,R} \); moreover, if \( n < m = -|R_T| \) or \( |R_T| = m < n \), then \( R'(a, u, n) \in C_{K,T,R} \) iff \( R'(a, u, m) \in C_{K,T,R} \).

The atoms \( R'(a, u, n) \) entailed by \( R(a, u, 0) \) in \( C_{K,T,R} \) via \((r1)-(r3)\), also have an impact, via \((ex)\), on the atoms of the form \( B(a, n) \) and \( B(u, n) \) in \( C_{K,T,R} \). Thus, in Example 10, \( R(a, v, 0) \) entails \( \exists Q(a, u) \), for \( n > 0 \). To analyse the behaviour of such atoms, it is helpful to assume that \( T \) is in concept normal form (CoNF) in the following sense: for every role \( R \in R_T \), the TBox \( T \) contains

\[
\exists R \subseteq A_R^0,
\phi_v \exists R \subseteq A_R^{-1},
\phi_v A_R^m \subseteq A_R^{-m},
\phi_v A_R^{-m} \subseteq A_R^m,
\phi_v A_R^m \subseteq A_R^{-m+1},
\phi_v A_R^{-m+1} \subseteq A_R^m,
\]

for \( 0 \leq m \leq |R_T| \) and some concepts \( A_R^i \), and

\( A_R^m \subseteq \exists R' \), for \( |m| \leq |R_T| \) and \( R'(a, v, m) \in C_{K,T,R} \).

(In Example 10 \( C_K \) will contain the atoms \( A_R^i(a, n) \) and \( A_R^i(a, n + 1) \), for \( n \geq 1 \).) By Lemma 11, if \( T \) is in CoNF, then we can compute the atoms \( B(a, n) \) and \( B(u, n) \) in \( C_{K,T,R} \) without using the rules \((r1)-(r3)\). Lemma 11 also implies that we can add the inclusions above (with fresh \( A_R^i \)) to \( T \) if required, thereby obtaining a conservative extension of \( T \); so from now on we always assume \( T \) to be in CoNF. These observations enable the proof of the following two lemmas. The first one characterises the atoms \( B(a, u) \) in \( C_{K,T,R} \):

**Lemma 12 (monotonicity)** Suppose \( a \leadsto_R^0 u \) in \( C_{K,T,R} \). If either \( m < n < 0 \) or \( 0 < n < m \), then \( B(a, n) \in C_{K,T,R} \) implies \( B(u, n) \in C_{K,T,R} \); moreover, if either \( n < m = -|T| \) or \( |T| = m < n \), then \( B(a, n) \in C_{K,T,R} \) iff \( B(u, m) \in C_{K,T,R} \).

The second lemma characterises the ABox part of \( C_K \) and is a straightforward generalisation of Lemma 7.

**Lemma 13** For any KB \( K = (\mathcal{T}, A) \) and any atom \( \alpha \) of the form \( A(a, n) \), \( \exists R(a, n) \) or \( R(a, b, n) \), where \( a, b \in \text{ind}(A) \) and \( n \in \mathbb{Z} \), we have \( \alpha \in C_K \) iff \( \alpha \in \text{cl}^{R_T}(A^2) \).

An obvious extension of the rewriting of Theorem 8 provides, for every CQ \( q(\bar{x}, \bar{s}) \), a UCQ \( \text{ext}_q^T(\bar{x}, \bar{s}) \) such that for all \( \bar{a} \subseteq \text{ind}(A) \) and \( \bar{n} \subseteq \mathbb{Z} \) of the appropriate length,

\[
C_K^0 \models q(\bar{a}, \bar{n}) \quad \text{iff} \quad A^2 \models \text{ext}_q^T(\bar{a}, \bar{n}).
\]
Moreover, for a basic concept $\exists R$, we find a UCQ $\text{ext}^T_{(R)}(\xi, \tau)$ such that, for any $a \in \text{ind}(A)$ and $n \in \mathbb{Z}$, $\exists R(a, n) \in C_K$ iff $A_a^n \models \text{ext}^T_{(R)}(a, n)$.

We now use the obtained results to show that one can find all answers to a TQL KB $K$ by only considering a fragment of $C_K$ whose size is polynomial in $|T|$ and $|q|$. This property is called the polynomial witness property [Gottlob and Schwentick, 2011]. Denote by $C_K^{d, \ell}$, for $d, \ell \geq 0$, the restriction of $C_K$ to the moments of time in the interval $[\min \text{tem}(A) - \ell, \max \text{tem}(A) + \ell]$.

Let $q(\vec{x}, \vec{s})$ be a TQL. Tuples $\vec{a} \subseteq \text{ind}(A)$ and $\vec{n} \subseteq \text{tem}(A)$ give a certain answer to $q(\vec{x}, \vec{s})$ over $K = (T, A)$ iff there is a homomorphism $h$ from $q$ to $C_K$, which maps individual (temporal) terms of $q$ to individual (respectively, temporal) terms of $C_K$ such that the following conditions hold:

1. $h(\vec{x}) = \vec{a}$ and $h(b) = b$, for all $b \in \text{ind}(A)$;
2. $h(\vec{s}) = \vec{n}$ and $h(m) = m$, for all $m \in \text{tem}(A)$;
3. $h(q) \subseteq C_K$.

where $h(q)$ denotes the set of atoms obtained by replacing every term in $q$ with its $h$-image, e.g., $P(\xi, \zeta, \tau)$ with $P(h(\xi), h(\zeta), h(\tau))$, $(\tau_1 < \tau_2)$ with $h(\tau_1) < h(\tau_2)$, etc.

Now, using the monotonicity lemmas for the temporal dimension and the fact that atoms of depth $> |R_T|$ in the canonical models duplicate atoms of smaller depth, we obtain

**Theorem 14** There are polynomials $f_1$ and $f_2$ such that, for any consistent TQL KB $K = (T, A)$, any CQ $q(\vec{x}, \vec{s})$ and any $\vec{a} \subseteq \text{ind}(A)$ and $\vec{n} \subseteq \text{tem}(A)$, we have $K \models q(\vec{a}, \vec{n})$ iff there is a homomorphism $h : q \rightarrow C_K$ such that $h(q) \subseteq C_K^{d, \ell}$, where $d = f_1(|T|, |q|)$ and $\ell = f_2(|T|, |q|)$.

We are now in a position to define a rewriting for any given CQ and TQL TBox.

## 5 UCQ Rewriting

Suppose $q(\vec{x}, \vec{s})$ is a CQ and $T$ a TQL TBox (in CoNF). Without loss of generality we assume $q$ to be totally ordered. By a sub-query of $q$ we understand any subset $q' \subseteq q$ containing all temporal constraints $(\tau < \tau')$ and $(\tau = \tau')$ that occur in $q$. In the rewriting for $q$ and $T$ given below, we consider all possible splittings of $q$ into two sub-queries (sharing the same temporal terms). One is to be mapped to the ABox part of the canonical model $C(T, A)$, and so we can rewrite it using (2). The other sub-query is to be mapped to the non-ABox part of $C(T, A)$ and requires a different rewriting.

For every $R \in R_T$, we construct the set $C_K^{d, \ell}_{T, R}$, where $d$ and $\ell$ are provided by Theorem 14. Let $h$ be a map from a sub-query $q_h \subseteq q$ to $C_K^{d, \ell}_{T, R}$ such that $h(q_h) \subseteq C_K^{d, \ell}_{T, R}$. Denote by $h^x_\ell$ the set of individual terms $\xi$ in $q_h$ with $h(\xi) = a$, and let $Y_h$ be the remaining set of individual terms in $q_h$. We call $h$ a witness for $R$ if

- $h^x_\ell$ contains at most one individual constant;
- every term in $Y_h$ is a quantified variable in $q$;
- $q_h$ contains all atoms in $q$ with a variable from $Y_h$.

Let $h$ be a witness for $R$. Denote by $\sim_R$ the union of all $\sim^n_R$ in $C_K^{d, \ell}_{T, R}$. Clearly, $\sim_R$ is a tree order on the individuals in $C_K^{d, \ell}_{T, R}$, with root $a$. Let $T_h$ be its minimal sub-tree containing $a$ and the $h$-images of all the individual terms in $q_h$. For each $v \in T_h \setminus \{a\}$, we take the (unique) moment $g(v)$ with $u \sim^n_R v$, for some $u$ and $R$, and set $g(a) = 0.$ For $A(y, \tau) \in q_h$, we say that $h(y)$ realises $A(y, \tau)$ for any $P(\xi, \xi', \tau) \in q_h$, there are $u, u' \in T_h$ with $u \sim_R u'$ and $\{u, u'\} = \{h(\xi), h(\xi')\}$; we say that $u'$ realises $P(\xi, \xi', \tau)$. Let $r$ be a list of fresh temporal variables $r_u$, for $u \in T_h \setminus \{a\}$. Consider the following formula, whose free variables are $r_u$ and the temporal variables of $q_h$:

$$t_h = \exists P \left( \bigwedge_{u \sim_R v} \delta^{g(v) - g(u)} (r_u, r_u) \land \bigwedge_{u \text{realises } h(\xi)} \delta^{h(\tau) - g(u)} (r_u, \tau) \right),$$
where the formulas $\delta^n(t, s)$ say that $t$ is at least $n$ moments before $s$: that is, $\delta^0(t, s)$ is $(t = s)$ and $\delta^n(t, s)$ is
\[
\exists s_1, \ldots, s_{n-1} (t < s_1 < \cdots < s_{n-1} < s), \quad \text{if } n > 0,
\]
\[
\exists s_1, \ldots, s_{[n]-1}(t > s_1 > \cdots > s_{[n]-1} > s), \quad \text{if } n < 0.
\]

Take a fresh variable $x_h$ and associate with $h$ the formula
\[
w_h = \exists r_a \exists x_h \left[ \exists T_R(x_h, r_a) \land \bigwedge_{h(\xi)=a} (\xi = x_h) \land t_h \right].
\]

To give the intuition behind $w_h$, suppose that $C_{(T, A)} \models g$ and $w_h$, for some assignment $g$. Then $g$ maps all terms in $A_h$ to $g(x_h) \in \text{ind}(A)$ such that $\exists R(g(x_h), g(r_n)) \in C_{(T, A)}$, so $(g(x_h), g(r_n))$ is the root of a substructure of $C_{(T, A)}$ isomorphic to $C_{k,T,h}$ in which the variables from $\gamma_h$ can be mapped according to $h$. For temporal terms, the formula $t_h$ cannot specify the values prescribed by $h$: without $\neg$ in UCQs, we can only say that $\tau$ is at least (not exactly) $n$ moments before $\tau'$. However, by Lemmas 11 and 12 this is still enough to ensure that $g$ and $h$ give a homomorphism from $q_h$ to $C_{(T, A)}$.

**Example 15** Let $T$ be the same as in Example 10 and let
\[
q(x, t) = \exists y, z, t' \left( (t < t') \land Q(x, y, t) \land S'(y, z, t') \right).
\]

The map $h = \{ x \mapsto a, y \mapsto v, z \mapsto u_1, t \mapsto 1, t' \mapsto 2 \}$ is a witness for $R$, with $q_h = q$ and $w_h$ is the following formula
\[
\exists r_a \exists x_h \left( \exists T_R(x_h, r_a) \land (x_h = x) \land \exists r_v \exists r_u_1 (\delta^0(r_a, r_v) \land \delta^1(r_v, r_u_1) \land \delta^1(r_u_1, t) \land \delta^1(r_u_1, t')) \right).
\]

We can now define a rewriting for $q(\vec{x}, \vec{s})$ and $T$. Let $\Sigma$ be the set of all witnesses for $q$ and $T$. We call a subset $\Theta \subseteq \Sigma$ consistent if $(X_{h_1} \cup Y_{h_1}) \cap (X_{h_2} \cup Y_{h_2}) \subseteq X_{h_1} \cap X_{h_2}$, for any distinct $h_1, h_2 \in \Theta$. Assuming that $\vec{y}$ contains all the quantified variables in $q$ and $q \setminus \Theta$ is the sub-query of $q$ obtained by removing the atoms in $q_h$, $h \in \Theta$, other than $(\tau < \tau')$ and $(\tau = \tau')$, we set:
\[
q^*(\vec{x}, \vec{s}) = \exists \vec{y} \bigvee_{\Theta \subseteq \Sigma} \left( \bigwedge_{h \in \Theta} w_h \land \exists T_{q(\vec{y}, \Theta)} \right).
\]

**Theorem 16** Let $T$ be a TQL TBox in CoNF and $q(\vec{x}, \vec{s})$ a totally ordered CQ. Then, for any consistent KB $(T, A)$ and any tuples $\vec{a} \subseteq \text{ind}(A)$ and $\vec{n} \subseteq \mathbb{Z}$,
\[
(T, A) \models q(\vec{a}, \vec{n}) \iff A^{\vec{a}} \models q^*(\vec{a}, \vec{n}).
\]

$(T, A)$ is inconsistent iff $(T^+, A) \models q^\bot$.

Theorem 2 now follows by Lemma 3.

## 6 Non-Rewritability

In this section, we show that the language TQL is nearly optimal as far as rewritability of CQs and ontologies is concerned.

We note first, that the syntax of TQL allows concept inclusions and role inclusions; ‘mixed’ axioms such as the datalog rule $A(x, t) \land R(x, y, t) \rightarrow B(x, t)$ are not expressible. The reason is that mixed rules often lead to non-rewritability, as is well known from the DL $\mathcal{EL}$. For example, there does not exist an FO-query $q(x, t)$ such that $(T, A) \models A(a, n)$ if $A^{\vec{a}} \models q(a, n)$ for $T = \{ A(y, t) \land R(x, y, t) \rightarrow A(x, t) \}$ since such a query has to express that at time-point $t$ there is an $R$-path from $x$ to some $y$ with $A(y, t)$.

Second, it would seem to be natural to extend TQL with the temporal next/previous-time operators as concept or role constructs. However, again this would lead to non-rewritability: any FO-rewriting for
A(x, t) and \{\Box y A \sqsubseteq B, \Box y B \sqsubseteq A\} has to express that there exists \(n \geq 0\) such that \(A(x, t - 2n)\) or \(B(x, t - (2n + 1))\), which is impossible [Libkin, 2004].

Another natural extension would be inclusions of the form \(A \sqsubseteq B\). (Note that inclusions of the form \(A \sqsubseteq \exists R.B\) are expressible in OWL 2 QL.) But again such an extension would ruin rewritability. The reason is that temporal precedence < is a total order, and so one can construct an ABox \(A\) and a UCQ \(q(x) = q_1 \lor q_2\) such that \((T, A) \models q(a)\) but \((T, A) \not\models q_i(a), i = 1, 2,\) for \(T = \{A \sqsubseteq \Diamond y B\}\). Indeed, we take \(A = \{A(a, 0), C(a, 1)\}\) and

\[
q_1(x) = \exists t (C(x, t) \land B(x, t)),
q_2(x) = \exists t, t' ((t < t') \land C(x, t) \land B(x, t')).
\]

In fact, by reduction of 2+2-SAT [Schaerf, 1993], we prove the following:

**Theorem 17** Answering CQs over the TBox \(\{A \sqsubseteq \Diamond y B\}\) is \(\text{CONP}\)-hard for data complexity.

### 7 Related Work

The Semantic Web community has developed a variety of extensions of RDF/S and OWL with validity time [Motik, 2012; Pugliese et al., 2008; Gutierrez et al., 2007]. The focus of this line of research is on representing and querying time stamped RDF triples or OWL axioms. In contrast, in our language only instance data are time stamped, while the ontology formulates time independent constraints that describe how the extensions of concepts and roles can change over time. In the temporal DL literature, a similar distinction has been discussed as the difference between temporalised axioms and temporalised concepts/roles; the expressive power of the respective languages is incomparable [Gabbay et al., 2003; Baader et al., 2012].

In Theorem 8 we show rewritability using boundedness of recursion. This connection between first-order definability and boundedness is well known from the datalog and logic literature where boundedness has been investigated extensively [Gaifman et al., 1987; van der Meyden, 2000; Kreutzer et al., 2007]. [2009] investigate boundedness for datalog programs on linear orders; the results are different from ours since the linear order is the only predicate symbol of the datalog programs considered and no further restrictions (comparable to ours) are imposed.

### 8 Conclusion

In this paper, we have proved UCQ rewritability for conjunctive queries and TQL ontologies over data instances with validity time. Our focus was solely on the existence of rewritings, and we did not consider efficiency issues such as finding shortest rewritings, using temporal intervals in the data representation or mappings between temporal databases and ontologies. We only note here that these issues are of practical importance and will be addressed in future work. It would also be of interest to investigate the possibilities to increase the expressive power of both ontology and query language. For example, we believe that the extension of TQL with the next/previous time operators, which can only occur in TBox axioms not involved in cycles, will still enjoy rewritability. We can also increase the expressivity of conjunctive queries by allowing the arithmetic operations \(+\) and \(\times\) over temporal terms, which would make the CQ \(A(x, t)\) and the TBox \(\{\Box y A \sqsubseteq B, \Box y B \sqsubseteq A\}\) rewritable in the extended language.
References


A Proofs

We first give a detailed definition of the standard translation $C^1$ and $S^2$ of TQL concepts $C$ and roles $S$ into two-sorted first-order logic. The definitions are by induction as follows. For concepts:

$$A^1(\xi, \tau) = A(\xi, \tau),$$
$$\bot^1(\xi, \tau) = \bot,$$
$$\exists R^1(\xi, \tau) = (\exists y \ R^1(\xi, y, \tau)),$$
$$C_1 \cap C_2^1(\xi, \tau) = C_1^1(\xi, \tau) \cap C_2^1(\xi, \tau),$$
$$\langle \rho, C \rangle^1(\xi, \tau) = \exists t ( (t < \tau) \land C^2(\xi, t)),$$
$$\langle \rho, C \rangle^1(\xi, \tau) = \exists t ( (t > \tau) \land C^2(\xi, t)).$$

For roles:

$$P^1(\xi, \zeta, \tau) = P(\xi, \zeta, \tau),$$
$$\langle P \rangle^1(\xi, \zeta, \tau) = P(\xi, \zeta, \tau),$$
$$\bot^1(\xi, \zeta, \tau) = \bot,$$
$$\langle S_1 \cap S_2 \rangle^1(\xi, \zeta, \tau) = S_1^1(\xi, \zeta, \tau) \land S_2^1(\xi, \zeta, \tau),$$
$$\langle \rho, S \rangle^1(\xi, \zeta, \tau) = \exists t ( (t < \tau) \land S^2(\xi, \zeta, t)),$$
$$\langle \rho, S \rangle^1(\xi, \zeta, \tau) = \exists t ( (t > \tau) \land S^2(\xi, \zeta, t)).$$

Lemma 1 Suppose $T$ is a flat TBox, let $n_T = (4 \cdot |T|)^4$. Then $cl^\infty(A^2) = cl^{n_T}(A^2)$, for any ABox $A$.

Proof. We start with an observation that to compute $cl^\infty(S)$ it is sufficient to first compute the closure under the rules (r1)–(r3) for role inclusions, then apply the rule (ex), and then apply the rules (c1)–(c3) for concept inclusions. Formally, for a set $R$ of rules, let $cl_R(S)$ denote the result of applying the rules in $R$ (non-recursively!) to $S$. Let role $= \{(r1), (r2), (r3)\}$ and concept $= \{(c1), (c2), (c3)\}$. Then

Fact 1. $cl^\infty(S) = cl^\infty_{concept}(cl^\infty_{(ex)}(cl^\infty_{role}(S)))$, for all $S$.

The proof is straightforward: since none of the rules (c1)–(c3) or (ex) introduces a new role assertion, i.e., an assertion of the form $R(u, v, n)$, no new applications of rules (r1)–(r3) become possible after applying rules (c1)–(c3) and (ex); and no new applications of (ex) becomes possible after applying rules (c1)–(c3).

We first consider the closure under the rules for role inclusions and show $cl^\infty_{role}(A^2) = cl^\infty_{role}(A^2)$ for $k_\rho = 4|R_T|^4$. We start with the observation that it is sufficient to show this for ABoxes having at most two individuals because role assertions for individuals $u, v$ do not interact with role assertions for individuals $u', v'$ if $\{u, v\} \neq \{u', v'\}$. Formally, for any $u, v$ ($u = v$ is not excluded), let $A_{u,v}$ consist of all assertions $R(u, v, n)$ in $A$. Then

Fact 2. $cl^k_{role}(A^2) = \bigcup_{u, v \in \Ind(A)} cl^k_{role}(A^2_{u,v})$, for all $k \geq 0$.

Now let $A$ be an ABox with individuals $u, v$. Observe that the rule (r1) is local in the sense that the addition of a role assertion at time point $n$ depends only on role assertions that hold already at time point $n$. It follows that $cl^\infty_{\{r1\}}(S) = cl^{|R_T|}(S)$. We now analyse the two operators obtained by adding to (r1) either the rule (r2) or the rule (r3). Let $P = \{(r1), (r2)\}$ and $F = \{(r1), (r3)\}$. For the rules in $P$ the addition of role assertions at time point $n$ only depends on the time points $m \leq n$ and, similarly, for $F$ the addition of role assertions at time point $n$ only depends on time points $m \geq n$. It is now easy to see that in each case, one has to alternate between applications of local rules and the rule (r2) (respectively (r3)) at most $|R_T|$ times. Thus

Fact 3. $cl^\infty_P(A^2) = cl^{|R_T|}_P(A^2)$ and $cl^\infty_F(A^2) = cl^{|R_T|}_F(A^2)$.

By Fact 3, to obtain a $k_\rho$ such that $cl^\infty_{role}(A^2) = cl^k_{role}(A^2)$ it is sufficient to determine an upper bound for the number of alternations between $cl^\infty_P$ and $cl^\infty_F$ that are required to compute $cl^\infty_{role}$.
Fact 4. \( \text{cl}^\infty_{R}(S) = (\text{cl}_{R}^{c} \circ \text{cl}_{R}^{c})^{[R,T]}(S) \).

To prove Fact 4 we introduce the notion of a cross over. Assume \( u, v \) are the individuals of \( S \). Let \( R_{1} \) and \( R_{2} \) be roles. We say that \((R_{1}, R_{2})\) are a cross over in \( S \) if there are \( m_{1}, m_{2} \) with \( m_{1} + 1 \geq m_{2} \) such that \( R_{1}(u, v, n) \in S \) for all \( n \leq m_{1} \) and \( R_{2}(u, v, n) \in S \) for all \( n \geq m_{2} \).

Claim 1. Let \( S = \text{cl}_{R}^{c}(S) \), \( S_{1} = \text{cl}_{R}^{c}(S) \) and \( S_{2} = \text{cl}_{R}^{c}(S_{1}) \). If \( S_{2} \supseteq S_{1} \) then there exists a cross over \((R_{1}, R_{2})\) in \( S_{2} \) which is not a cross over in \( S \).

Proof of Claim 1. Since \( S_{1} \) is closed under (r1), there exist \( \emptyset, R \subseteq R' \) in \( T \) and \( n_{1} \) such that

1. \(- R(u, v, n_{1}) \in S_{1} \) and there is \( n > n_{1} \) with \( R'(u, v, n_{1}) \notin S_{1} \);  
2. and \( R'(u, v, n) \in S_{2}, \) for all \( n > n_{1} \).

It follows that \( R(u, v, n_{1}) \notin S \); for otherwise \( R'(u, v, m) \in S_{1} \) for all \( n > n_{1} \). From \( R(u, v, n_{1}) \in S_{1} \) and, since \( S \) is closed under (r1), there exist \( \emptyset, S \subseteq S' \) in \( T \) and \( n_{2} > n_{1} \) such that

1. \(- S(u, v, n_{2}) \in S \) and \( S'(u, v, n_{1}) \notin S \);  
2. \( S'(u, v, n) \in S_{1}, \) for all \( n < n_{2} \).

Let \( m_{2} = n_{2} - 1, m_{2} = n_{1} + 1 \). Then \((S', R')\) is a cross over in \( S_{2} \) with witness times points \( m_{1}, m_{2} \) which is not a cross over in \( S \). This finishes the proof of Claim 1.

Clearly the number of cross overs is bounded by \(|R_{T}|^{2}\) and so we have proved Fact 4. We obtain from Fact 3 and Fact 4 that \( \text{cl}^\infty_{R}(A^{2}) = \text{cl}^{k}_{R}(A^{2}) \) for \( k_{c} = 4|R_{T}|^{4} \).

One can show in almost exactly the same way that \( \text{cl}^\infty_{concept}(A^{2}) = \text{cl}^{k}_{concept}(A^{2}) \) for \( k_{c} = 4m_{c}^{2} \), where \( m_{c} \) is the number of concept names in \( T \). Since \( 4|R_{T}|^{4} + 4m_{c}^{2} \leq (4|T|)^{4} \), this finishes the proof of Lemma 4. \( \square \)

Lemma [11] Let \( a \sim_{R}^{0} u \) in \( C_{K,T} \). Then the following hold, for all basic roles \( R' \):

1. if \( m < n < 0 \) or \( 0 \leq n < m \), then \( R'(a, u, n) \in C_{K,T} \) implies \( R'(a, u, m) \in C_{K,T} \);
2. if \( n < m = -|R_{T}| \) or \( |R_{T}| = m < n \), then \( R'(a, u, n) \in C_{K,T} \) iff \( R'(a, u, m) \in C_{K,T} \).

Proof. We start with Item 1. Assume \( 0 < n < m \) (the case \( m < n < 0 \) is similar and left to the reader). The proof is by induction over rule applications. Namely, we show

Claim 1. Let \( 0 < n < m \). If \( R'(a, u, n) \in \text{cl}^{k}_{1}(A^{2}) \), then \( R'(a, u, m) \in \text{cl}^{k}_{1}(A^{2}) \), for all \( k \geq 0 \).

For \( A^{2} \) itself the claim is trivial. Now assume it holds for \( \text{cl}^{k}_{1}(A^{2}) \). Applications of (ex) and (c1)–(c3) do not influence the role assertions for \( (a, u) \), so we do not have to consider them. Applications of (r1)–(r3) clearly preserve the property stated in Claim 1.

Now consider Item 2. Assume \( |R_{T}| = m < n \) (the case \( -|R_{T}| = m > n \) is similar and left to the reader). The proof is by induction over rule applications. In detail, we show the following

Claim 2. For all \( k \geq 0 \) and \( \ell > 1 \), if there exists \( R' \) with \( R'(a, u, \ell) \in \text{cl}^{k}_{1}(A^{2}) \) and \( R'(a, u, \ell - 1) \notin \text{cl}^{k}_{1}(A^{2}) \), then \( |R''| \circ R' \subseteq \text{cl}^{k}_{1}(A^{2}) \geq \ell \).

The proof of Claim 2 is by induction over \( k \) and left to the reader. Now Item 2 follows directly with Point 1. \( \square \)

We now analyse in more detail why one can without loss of generality assume TBoxes to be in CoNF. Let \( \mathcal{T} \) be a TBox in normal form. Add to \( \mathcal{T} \) the inclusions

\[
\exists R \subseteq A_{R}^{0}, \quad \Diamond_{F} \exists R \subseteq A_{R}^{-1}, \quad \Diamond_{F} A_{R}^{-m} \subseteq A_{R}^{-m-1}, \quad (3) \\
\Diamond_{P} \exists R \subseteq A_{R}^{1}, \quad \Diamond_{P} A_{R}^{m} \subseteq A_{R}^{m+1}. \quad (4)
\]
for all $R \in R_T$ and $0 \leq m \leq |R_T|$, where the $A^i_R$ are fresh concept names; and add

$$A^0_R \subseteq \exists R', \text{ for } |m| \leq |R_T| \text{ and } R'(a, v, m) \in C_{K_{T,R}}. \quad (5)$$

Denote the resulting TBox by $\mathcal{T}'$. We first show that $\mathcal{T}'$ is a conservative extension of $\mathcal{T}$ in the following sense:

**Lemma 18** For every model $\mathcal{I}$ of $\mathcal{T}$ there exists a model $\mathcal{I}'$ of $\mathcal{T}'$ such that $\Delta^I = \Delta^{I'}$ and such that $\mathcal{I}'$ coincides with $\mathcal{I}$ for the interpretation of symbols from $\mathcal{T}$.

**Proof.** Assume $\mathcal{I}$ is given. We define $\mathcal{I}'$ as follows:

$$A^n_R(\mathcal{I}(n)) = (\exists R)(\mathcal{I}(n)), \quad (A^m_R)^+(\mathcal{I}(n)) = (\forall p \exists R)(\mathcal{I}(n)) \quad \text{and} \quad (A^m_R)^-(\mathcal{I}(n)) = (\forall p \forall R)(\mathcal{I}(n)), \quad \text{for } 0 < m \leq |R_T|.$$ 

We have to show that $\mathcal{I}' \models \mathcal{T}'$. It is readily seen that $\mathcal{I}'$ satisfies the inclusions (3) and (4), for all $R \in R_T$ and $0 \leq m \leq |R_T|$. The interesting part are the fresh inclusions $A^0_R \subseteq \exists R'$ for $R'(a, v, m) \in C_{K_{T,R}}$. Let $A^0_R \subseteq \exists R'$ be such a fresh inclusion. Consider $m \geq 0$ and let $d \in (A^m_R)(\mathcal{I}(n))$. Then $d \in (\forall p \exists R')(\mathcal{I}(n))$. Moreover, $R'(a, v, m) \in C_{K_{T,R}}$ implies, by Lemma 11, $T |\models \forall p R \subseteq R'$. Since $\mathcal{I}$ is a model of $\mathcal{T}$, we obtain $d \in (\exists R')(\mathcal{I}(n))$. \hfill \Box

It also follows from Lemma 18 that the set of role assertions in $C_{K_{T,R}}$ coincides with the set of role assertions in $C_{K_{T,A}}$ and so, $\mathcal{T}'$ is in CoNF.

Now observe that if a TBox $\mathcal{T}$ is in CoNF, then one can construct $\mathcal{T}_{T,A}$ by

- first applying the rules (r1)–(r3) exhaustively to ABox individuals,
- then applying the rules (ex), (−) and (c1)–(c3) exhaustively,
- and finally applying again (r1)–(r3).

This follows from the second part of Lemma 11 (according to which the role assertions are stable at distances larger than $|R_T|$ in $C_{K_{T,A}}$) and the inclusions (5).

**Lemma 12** Let $\mathcal{T}$ be in CoNF and a $\sim^0_R \alpha$ in $C_{K_{T,R}}$. Then the following hold, for all basic concepts $B$:

- if $m < n < 0$ or $0 < n < m$, then $B(u, n) \in C_{K_{T,R}}$ implies $B(u, m) \in C_{K_{T,R}}$;
- if $m < n = -|\mathcal{T}|$ or $|\mathcal{T}| = m < n$, then $B(u, n) \in C_{K_{T,R}}$ iff $B(u, m) \in C_{K_{T,R}}$.

**Proof.** The proof is similar to the proof of Lemma 11 and omitted. \hfill \Box

To generalize the rewriting from flat TBoxes to arbitrary TBoxes, we admit PEQs that can contain atoms of the form $\exists R(\xi, \tau)$, where $\exists R$ is a basic concept. Moreover, we modify the standard translation $C^\mathcal{T}$ to a translation $C^\mathcal{T}$ that regards basic concepts $\exists R$ as atoms by setting $(\exists R)^{\mathcal{T}}(\xi, \tau) = \exists R(\xi, \tau)$. We now show how the definition of $\varphi^{n+1}$ is modified. Given a generalized PEQ $\varphi$, we set $\varphi^{0} = \varphi$ and define, inductively, $\varphi^{(n+1)}$ as the result of replacing

- every $A(\xi, \tau)$ in $\varphi$ with $A(\xi, \tau) \lor \bigvee_{C \subseteq A \in \mathcal{T}} (C^{\mathcal{T}}(\xi, \tau))^{n+1}$,
- every $P(\xi, \zeta, \tau)$ in $\varphi$ with $P(\xi, \zeta, \tau) \lor \bigvee_{S \subseteq P \in \mathcal{T}} (S^{\mathcal{T}}(\xi, \zeta, \tau))^{n+1}$,
- every $\exists R(\xi, \tau)$ in $\varphi$ with $\exists y R^{\mathcal{T}}(\xi, \zeta, \tau) \lor \bigvee_{S \subseteq R \in \mathcal{T}} (S^{\mathcal{T}}(\xi, \zeta, \tau))^{n+1}$ \lor $\bigvee_{C \subseteq \exists R \in \mathcal{T}} (C^{\mathcal{T}}(\xi, \tau))^{n+1}$.  

15
For a query \( q(\bar{x}, \bar{s}) \) we now define as \( \text{ext}_T^q(\bar{x}, \bar{s}) \) the result of replacing every atom \( A(\xi, \tau) \) by \( (A(\xi, \tau))^{n \tau} \) and every atom \( P(\zeta, \zeta, \tau) \) by \( (P(\zeta, \zeta, \tau))^{n \tau} \) and replacing in the resulting PEQ the atoms \( (\exists \bar{R})(\xi, \tau) \) by \( (\exists y \bar{R}^y(\xi, y, \tau)) \). One can readily show that

\[
\text{C}_T^{\alpha} \models q(\bar{a}, \bar{n}) \iff \text{A}_{\text{ext}}^{\alpha} \models \text{ext}_T^q(\bar{a}, \bar{n}).
\]

We also set \( \text{ext}_{\exists \bar{R}}^T(\xi, \tau) = (\exists \bar{R}(\xi, \tau))^{n \tau} \). Then one can show

\[
\exists \bar{R}(a, n) \in C_K \iff \text{A}_{\text{ext}}^{\alpha} \models \text{ext}_{\exists \bar{R}}^T(a, n).
\]

To prove Theorem 14 we require some preparation. Firstly, variations of the monotonicity Lemmas 11 and 12 can be proved for ABox individuals in arbitrary ABoxes as well.

**Lemma 19** For any \( K = (T, A) \) with \( T \) in CoNF the following hold:

- if \( m < n < \min \text{tem}(A) \) or \( \max \text{tem}(A) < n < m \), then
  \[
  R(a, b, n) \in C_K \implies \exists R(a, b, m) \in C_K, \quad \text{for all basic roles } R \text{ and any } a, b \in \text{ind}(A),
  \]
  \[
  B(a, n) \in C_K \implies \exists B(a, m) \in C_K, \quad \text{for all basic concepts } B \text{ and any } a \in \text{ind}(A);
  \]

- if \( n < m = \min \text{tem}(A) - |R_T| \) or \( \max \text{tem}(A) + |R_T| = m < n \), then
  \[
  R(a, b, n) \in C_K \iff \exists R(a, b, m) \in C_K\quad \text{quad for all basic roles } R \text{ and any } a, b \in \text{ind}(A),
  \]
  \[
  B(a, n) \in C_K \iff \exists B(a, m) \in C_K, \quad \text{for all basic concepts } B \text{ and any } a \in \text{ind}(A).
  \]

**Proof.** The proof is similar to the proof of Lemma 11 and omitted. \( \square \)

In what follows it will often be useful to work with types. Given \( (u, n) \) we denote by \( t(u, n) \) the set of basic concepts \( B \) with \( B(u, n) \in C_K \) and given \( (u, n), (v, n) \) we denote by \( t(u, v, n) \) the set of basic roles \( R \) with \( R(u, v, n) \in C_K \), where the knowledge base \( K \) will always be clear from the context.

Secondly, it will be useful to introduce a notation system for the individuals \( u \) and pairs \( (u, n) \) in \( C_K \). In detail, we identify any \( u \) in \( C_K \) with a vector

\[
(a, n_0, R_0, n_1, \ldots, n_k, R_k)
\]

which is defined inductively as follows. If \( u \) has depth 0, then \( u = a \in \text{ind}(A) \) and we denote \( u \) by the singleton vector \( (a) \). If \( u \) has depth \( k + 1 \) then there is a unique \( v \) of depth \( k \) and \( v \to^*_R u \). So, if

\[
v = (a, n_0, R_0, \ldots, n_k, R_k)
\]

then we set

\[
u = (a, n_0, R_0, \ldots, n_k, R_k, n_{k+1}, R_{k+1}),
\]

where \( R_{k+1} = R \) and \( n_{k+1} = n - (n_0 + \cdots + n_k) \). Moreover, if \( u = (a, n_0, R_0, \ldots, n_k, R_k) \), then the pair \( (u, n) \) is identified with

\[
(a, n_0, R_0, n_1, \ldots, n_k, R_k, n_{k+1})
\]

where \( n_{k+1} = n - (n_0 + \cdots + n_k) \). Observe that we can recover the time point \( n \) of any pair \( (u, n) \) as \( n = n_0 + \cdots + n_{k+1} \).

Observe that not every vector of this form is identical to some individual \( u \) in \( C_K \). It is easy to see which ones are, however: \( (a, n_0, R_0, n_1, \ldots, R_k) \) is identical to some \( u \) in \( C_K \) iff, inductively,

\[
a \in \text{ind}(A) \quad \text{and} \quad \exists R_i \in t(a, n_0, R_0, \ldots, n_i), \quad \text{for all } 0 \leq i \leq k.
\]

**Lemma 20** For any \( u \in C_{K_{\tau,n}} \), there is \( v \) of depth \( \leq 2|R_T| \) such that \( B(u, k) \in C_{K_{\tau,n}} \) implies \( B(v, k) \in C_{K_{\tau,n}} \) for all \( k \in \mathbb{Z} \) and all basic concepts \( B \).
Theorem 14. There are polynomials $f_1$ and $f_2$ such that, for any consistent TQL KB $K = (\mathcal{T}, \mathcal{A})$, any CQ $q(\vec{x}, \vec{s})$ and any $\vec{a} \subseteq \text{ind}(\mathcal{A})$ and $\vec{n} \subseteq \text{tem}(\mathcal{A})$, we have $K \models q(\vec{a}, \vec{n})$ if there is a homomorphism $h: q \to K$ such that $h(\vec{q}) \subseteq C^\ell_K$ and $\ell = f_2(|\mathcal{T}|, |q|)$. We set $d = 2|R_T| + |q|$.

Proof. Assume that a homomorphism $h: q \to K$ is given. A sub-query of $q$ is a non-empty subset $q' \subseteq q$ containing all temporal atoms in $q$. We assume that $q$ is totally ordered. First we consider the parameter $d$ for the depth required to find a match for $q$ in $C^d_K$. We set $d = 2|R_T| + |q|$.

A sub-query $q'$ of $q$ is connected if for any two individual terms $\xi_1, \xi_2$ in $q'$ there are $\tau_1, \ldots, \tau_n$ and $\zeta_0, \ldots, \zeta_n$ such that there are roles $R_1, \ldots, R_n$ with $R_i(\zeta_0, \zeta_1, \tau_1), \ldots, R_n(\zeta_{n-1}, \zeta_n, \tau_n) \in q'$ such that $\zeta_0 = \xi_1$ and $\zeta_n = \xi_2$. We consider the maximal connected components $q_1, \ldots, q_n$ of $q$ and construct the new homomorphism $h'$ as follows:

- $h'(\tau) = h(\tau)$ for all temporal terms $\tau$;
- For all $q_i$ such that $h(\xi) \in \text{ind}(\mathcal{A})$ for some $\xi$ in $q_i$, we have $h(y) \in C_{K}^{2|q_i|}$ for all $y$ in $q_i$. Thus, for individual terms $\xi$ in such a $q_i$ we set $h'(\xi) = h(\xi)$.
- Let $q_i$ be such that $h(\xi) \not\in C_{K}^{2|R_T|+|q|}$ for some $\xi$ in $q_i$. Take $\xi$ from $q_i$ such that $h(\xi)$ is of minimal depth in $C_{K}$, which, by our assumption, exceeds $2|R_T|$. By Lemma 20, we find $\nu$ of depth $\leq 2|R_T|$ such that $t(h(\xi), \nu) \subseteq t(v, m)$ for all $m \in \mathbb{Z}$. Assume $v = (a, n_0, R_0, \ldots, n_k, R_k)$ and $h(\xi) = (b, m_0, R'_0, \ldots, m_r, R'_r)$.
Let $\xi'$ be an individual variable in $q$. Then $h(\xi')$ is the concatenation of $h(\xi)$ and some vector

$$(m_{r+1}, R'_{r+1}, m_{r+2}, \ldots, m_s, R'_s).$$

Denote $m = (m_0 + \cdots + m_r) - (n_0 + \cdots + n_k)$ and define

$$h'(\xi') = (a, n_0, \ldots, n_k, R_k, m_{r+1} + m, R'_{r+1}, m_{r+2}, \ldots, m_s, R'_s).$$

The resulting $h'$ is the required homomorphism.

Now we consider the bound $\ell$ for the “width” of the match for $q$. Assume that $h: q \rightarrow C^d_k$ for $d = 2|R_T| + |q|$. We set $\ell = |T| \cdot |q| \cdot d$ and transform $h$ into a homomorphism $h': q \rightarrow C^{d, \ell}_k$. It should be clear that the required polynomials are $f_1(|T|, |q|) = 2|R_T| + |q|$ and $f_2(|T|, |q|) = |T| \cdot |q| \cdot f_1(|T|, |q|)$, respectively.

We define the **temporal extension $u'$** of an individual $u$ with representation $(a, n_0, R_0, \ldots, R_k)$ as the set $\{\overline{n_0}, \overline{n_1}, \ldots, \overline{n_k}\}$, where

$$\overline{n_k} = n_0 + \cdots + n_k.$$ 

By $\mathcal{H}^t_k$ we denote the set of all $h(\tau)$ and all $(h(\xi))'$, with $h(\xi)$ are represented as vectors as introduced above. Let $\mathcal{M} = \{\min \text{tem}(A) - |T|, \max \text{tem}(A) + |T|\}$. The homomorphism $h'$ we are going to construct will have the following property:

- $|m_1 - m_2| \leq |T|$, for any two $m_1, m_2 \in \mathcal{H}^t_k \cup \mathcal{M}$ such that there is no $m \in \mathcal{H}^t_k$, between $m_1$ and $m_2$, and such that $m_1, m_2 \leq \min \text{tem}(A) - |T|$ or $m_1, m_2 \geq \max \text{tem}(A) + |T|$.

Assume such an $h'$ has been constructed. The cardinality of $\mathcal{H}^t_k$, is bounded by $|\mathcal{M}| \cdot d$ (since $h'$ is into $C^{d, \ell}_k$). Hence $h'$ is into $C^{d, \ell}_k$, as required.

For the construction of $h'$, let $m_1, m_2 \in \mathcal{H}^t_k \cup \mathcal{M}$ be such that there is no $m \in \mathcal{H}^t_k$, between $m_1$ and $m_2$, and such that $m_1 < m_2 \leq \min \text{tem}(A) - |T|$ or $\max \text{tem}(A) + |T| \leq m_1 < m_2$. Assume $|m_2 - m_1| > |T|$ and that, without loss of generality, $\max \text{tem}(A) + |T| \leq m_1 < m_2$. We define $h'$ as follows. Let $m = (m_2 - m_1) - |T|$. Then, for all temporal terms $\tau$ we set

$$h'(\tau) = \begin{cases} h(\tau), & \text{if } h(\tau) \leq m_1, \\ h(\tau) - m, & \text{if } h(\tau) \geq m_2. \end{cases}$$

To define $h'(\xi)$ assume that $h(\xi) = (a, n_0, R_0, n_1, \ldots, R_k)$. If the temporal extension of $h(\xi)$ contains no time point $> m_1$, then $h'(\xi) = h(\xi)$. Otherwise,

- replace $n_0$ by $n_0 - m$ if $n_0 \geq m_2$,
- replace all $n_i$ by $n_i - m$ if $\overline{n}_{i-1} \leq m_1$ and $\overline{n}_i \geq m_2$,
- replace all $n_i$ by $n_i + m$ if $\overline{n}_{i-1} \geq m_2$ and $\overline{n}_i \leq m_1$

and let $h'(\xi)$ be the resulting vector. Using Lemmas 19, 11, and 12 one can readily check that $h'$ is a homomorphism.

After applying the above construction exhaustively, the resulting $h'$ is as required.

**Theorem 16** Let $T$ be a TQL TBox in CoNF and $q(\vec{x}, \vec{s})$ a totally ordered CQ. Then, for any consistent KB $(T, A)$ and any tuples $\vec{a} \subseteq \text{ind}(A)$ and $\vec{n} \subseteq \mathbb{Z}$,

$$(T, A) \models q(\vec{a}, \vec{n}) \iff A^{\vec{a}} \models q^*(\vec{a}, \vec{n}).$$

**Proof.** ($\Rightarrow$) Suppose $(T, A) \models q(\vec{a}, \vec{n})$. Then, by Theorem 14 there is a homomorphism $g: q(\vec{a}, \vec{n}) \rightarrow C^d_K$. If there is no individual variable $y$ such that $g(y) \notin \text{ind}(A)$, then we have $C^d_K \models q(\vec{a}, \vec{n})$, and so, by 2. $A^{\vec{a}} \models \text{ext}_q(\vec{a}, \vec{n})$, from which $A^{\vec{a}} \models q^*(\vec{a}, \vec{n})$ (just take $\emptyset = \emptyset$).
Otherwise, we take a variable $y$ with $g(y) \notin \text{ind}(\mathcal{A})$ and consider the minimal set $\Theta_y$ of atoms in $q$ with the following property: (i) all atoms containing $y$ are in $\Theta_y$; (ii) if $z$ is an individual variable in an atom from $\Theta_y$ and $g(z) \notin \text{ind}(\mathcal{A})$ then all atoms with $z$ are in $\Theta_y$. Consider the sub-query $q' \subseteq q$ comprised of all atoms in $\Theta_y$ and all the temporal constants in $q$. Denote by $X'$ the set of individual terms in $q'$ sent by $g$ to $\text{ind}(\mathcal{A})$, and by $\mathcal{Y}'$ the remaining individual terms in $q'$. Clearly, all elements of $\mathcal{Y}'$ are existentially quantified variables. In view of the UNA, $X'$ contains at most one constant. By the construction of $C_k$ and $\Theta_y$, there are unique $b \in \text{ind}(\mathcal{A})$, $n \in \mathbb{Z}$ and a role $R$ such that $b \sim_R^n u$, for some $u$, and, for every $z \in \mathcal{Y}'$, we have $b \sim_R^{n_i} u \sim_R^{n_1} \cdots \sim_R^{n_m} z$, for some $n_i$.

Consider now $C_{k,T,R}$ with root $a$. Let $e$ be the natural embedding of $C_{k,T,R} \setminus \{A(a,0)\}$ to $C_k$ such that $e(a) = b$, $e(0) = n$ and $e(R(a,v,0)) = R(b,u,n)$. Let $h$: $q' \rightarrow C_{k,T,R}$ be such that $h \circ e = g$ on $q'$. The map $h$ is not necessarily a tree witness for $R$ with $q_h = q'$; although $h(q') \subseteq C_{k,T,R}^d$, we may have $h(q') \nsubseteq C_{k,T,R}^f$. We use $h$ to construct a tree witness $h'$ for $R$ in the same way as in the proof of Theorem 14.

Let $\mathcal{S}$ be the set of all the distinct tree witnesses constructed in this way for the individual variables in $q$. Observe that if $y, z \in \mathcal{Y}'$, then $y$ and $z$ belong to the same tree witness. It follows that $\mathcal{S}$ is consistent. It follows now from (2), (6) and the proof of Theorem 14 that

$$A^\mathcal{S} \models g \bigwedge_{h' \in \mathcal{S}} h'^T \land \text{ext}_{q\setminus \mathcal{S}}^T.$$  

(⇐) Suppose $A^\mathcal{S} \models \neg g \bigwedge_{h' \in \mathcal{S}} h'^T \land \text{ext}_{q\setminus \mathcal{S}}^T$ for some consistent set $\mathcal{S}$ of tree witnesses and some assignment $g'$. Our aim is to extend $g'$ to an assignment $g$ in $C_k$ under which $C_k \models \neg q$. We set $g$ to coincide with $g'$ on those terms that occur in $q'$. Thus, it remains to define $g$ on the individual variables occurring in every $q_h$, $h \in \mathcal{S}$. Suppose $h$ is a tree witness for $R$, $g'(r_a) = v$ and $g'(x_b) = b$, where $r_a$ and $x_b$ are from $\mathcal{W}_h$. By (6), we have $C_k \models \exists R(b,u,n)$. Denote by $e$ the natural embedding of $C_{k,T,R}$ to $C_k$. Now, for every existentially quantified variable $y \in \mathcal{Y}_h$, we set $g(y) = h(e(y))$. Note that this definition is sound as different tree witnesses $h_1$ and $h_2$ in $\mathcal{S}$ do not share variables apart from $X_h \cap X_{h_2}$, and the variables in the $\mathcal{Y}_h$ do not occur in $q'$. That the resulting $g$ is a homomorphism from $q$ to $C_k$ follows from Lemmas 11 and 12 for the atoms in the tree witnesses $q_h$ and from (2) and (6) for the remaining atoms. □

**Theorem 17** Answering CQs over $\mathcal{T} = \{A \subseteq \ell, B\}$ is coNP-hard for data complexity.

**Proof.** The proof is by reduction of 2+2-SAT, a variant of propositional satisfiability that was first introduced by Schaefer as a tool for establishing lower bounds for the data complexity of query answering in a DL context [Schaefer, 1993]. A 2+2 clause is of the form ($p_1 \lor p_2 \lor \neg n_1 \lor \neg n_2$), where each of $p_1, p_2, n_1, n_2$ is a propositional variable or a truth constant $0$ and $1$. A 2+2 formula is a finite conjunction of 2+2 clauses. Now, 2+2-SAT is the problem of deciding whether a given 2+2 formula is satisfiable. 2+2-SAT is NP-complete [Schaefer, 1993].

Let $\varphi = c_0 \land \cdots \land c_n$ be a 2+2 formula in propositional variables $\pi_0, \dots, \pi_m$, and let $c_i = p_{i,1} \lor p_{i,2} \lor \neg n_{i,1} \lor \neg n_{i,2}$ for all $i \leq n$. Our aim is to define an ABox $\mathcal{A}_\varphi$ and a CQ $q$ such that $\varphi$ is unsatisfiable iff $(\mathcal{T}, \mathcal{A}_\varphi) \models q$. We expand $\mathcal{A}$ and $q_1 \lor q_2$ from Section 5. Namely, we represent the formula $\varphi$ in the ABox $\mathcal{A}_\varphi$ as follows:

- the individual name $f$ represents the formula $\varphi$ and the individual names $c_0, \ldots, c_n$ represent the clauses of $\varphi$;
- the assertions $S(f, c_0, 0), \ldots, S(f, c_n, 0)$ associate $f$ with its clauses (at time point 0), where $S$ is a role name;
- the individual names $\pi_0, \ldots, \pi_m$ represent the propositional variables, and the individual names false, true represent truth constants;
- the assertions, for each $i \leq n$,

$$P_1(c_i, p_{i,1}, 0), \quad P_2(c_i, p_{i,2}, 0), \quad N_1(c_i, n_{i,1}, 0), \quad N_2(c_i, n_{i,2}, 0)$$

associate each clause with the four variables/truth constants that occur in it (at time point 0), where $P_1, P_2, N_1, N_2$ are role names.
We further extend $A_\varphi$ to enforce a truth value for each of the variables $\pi_i$. To this end, add $A_0, \ldots, A_m$ to $A_\varphi$, where $A_i = \{A(a^i, 0), C(a^i, 1)\}$. Intuitively, $A_i$ is a copy of $A$ from Section 6 and is used to generate a truth value for the variable $\pi_i$, where we want to interpret $\pi_i$ as true if $q_1(a)$ is satisfied and as false if $q_2(a)$ is satisfied. To actually relate each individual name $\pi_i$ to the associated ABox $A_i$, we use role name $R$. More specifically, we extend $A_\varphi$ as follows:

- link variables $\pi_i$ to the ABoxes $A_i$ by adding assertions $R(\pi_i, a^i, 0)$ for all $i \leq m$; thus, truth of $\pi_i$ means that $tt(\pi_i)$ is satisfied and falsity means that $ff(\pi_i)$ is satisfied, where

\[
\begin{align*}
\tt(y) & = \exists z, t \left( R(y, z, 0) \land C(z, t) \land B(z, t) \right), \\
\ff(y) & = \exists z, t, t' \left( R(y, z, 0) \land (t < t') \land C(z, t) \land B(z, t') \right);
\end{align*}
\]

- to ensure that false and true have the expected truth values, add the following assertions to the ABox:

\[
\begin{align*}
R(\text{true}, \text{true}, 0), A(\text{true}, 0), C(\text{true}, 1), B(\text{true}, 1), \\
R(\text{false}, \text{false}, 0), A(\text{false}, 0), C(\text{false}, 1), B(\text{false}, 2).
\end{align*}
\]

Consider the query

\[
q = \exists x \left( S(f, x, 0) \land (\exists y_1 \left( P_1(x, y_1, 0) \land \ff(y_1) \right)) \land (\exists y_2 \left( P_2(x, y_2, 0) \land \ff(y_2) \right)) \land (\exists y_3 \left( N_1(x, y_3, 0) \land \tt(y_3) \right)) \land (\exists y_4 \left( N_2(x, y_4, 0) \land \tt(y_4) \right)) \right);
\]

which describes the existence of a clause with only false literals and thus captures falsity of $\varphi$. It is straightforward to show that $\varphi$ is unsatisfiable iff $(T, A_\varphi) \models q$. To obtain the desired CQ, it remains to pull the existential quantifiers out.

$\blacksquare$