Technische Universität Dresden
Institute for Theoretical Computer Science
Chair for Automata Theory

LTCS–Report

Temporal Query Answering in *DL-Lite* with Negation

Stefan Borgwardt Veronika Thost

LTCS-Report 15-16

Postal Address:
Lehrstuhl für Automatentheorie
Institut für Theoretische Informatik
TU Dresden
01062 Dresden

http://lat.inf.tu-dresden.de

Visiting Address:
Nöthnitzer Str. 46
Dresden

Abstract

Ontology-based query answering augments classical query answering in databases by adopting the open-world assumption and by including domain knowledge provided by an ontology. We investigate temporal query answering w.r.t. ontologies formulated in *DL-Lite*, a family of description logics that captures the conceptual features of relational databases and was tailored for efficient query answering. We consider a recently proposed temporal query language that combines conjunctive queries with the operators of propositional linear temporal logic (LTL). In particular, we consider negation in the ontology and query language, and study both data and combined complexity of query entailment.

Contents

1 Introduction

Ontologies play a central role in various applications: by linking data from heterogeneous sources to the concepts and relations described in an ontology, the integration and automated processing of the data can be considerably enhanced. In particular, queries formulated in the abstract vocabulary of the ontology can then be answered over all the linked datasets. Well-known medical domain ontologies like GALEN[1], for example, may capture the facts that the varicella zoster virus (VZV) is a virus, that chickenpox is a VZV infection, and that a negative allergy test implies that no allergies are present, by so-called *concept inclusions*: $\mathsf{VZV} \sqsubseteq \mathsf{Virus}, \mathsf{Chickenpox} \sqsubseteq \mathsf{VZVInfection}, \mathsf{NegAllergyTest} \sqsubseteq \neg\exists\mathsf{AllergyTo}$. Here, Virus is a *concept name* that represents the set of all viruses, and $\mathsf{AllergyTo}$ is a *role name*, i.e., a binary relation, which connects patients to allergies; $\exists\mathsf{AllergyTo}$ refers to the domain of this relation. A possible data source storing patient data (e.g., allergy test results and findings) could look as follows:

PID	Name
1	Ann
2	Bob
3	Chris

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

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

The data is connected to the ontology by mappings [PLC$^+$08], which in our example may link the tuple $(1, \text{Chickenpox}, 16.01.2011)$ to the facts $\mathsf{HasFinding}(1, \mathtt{x})$ and $\mathsf{Chickenpox}(\mathtt{x})$.

Ontology-based query answering (OBQA) over the above knowledge can, for example, assist in finding appropriate participants for a clinical study, by formulating the eligibility criteria as queries over the—usually linked and heterogeneous—patient data. The following are examples of in- and exclusion conditions for a recently proposed clinical trial:[2]

- The patient should have been previously infected with VZV or previously vaccinated with VZV vaccine.
- The patient should not be allergic to VZV vaccine.

Considering the first condition, OBQA would augment standard query answering (e.g., in SQL) w.r.t. the above ontology and data in that not only Bob and Chris but also Ann would be considered as an appropriate candidate. However, in standard OBQA, the queries neither allow for negation nor can refer to several

[1] `http://www.co-ode.org/ontologies/galen`

[2] `https://clinicaltrials.gov/ct2/show/NCT01953900`

points in time, both of which would be needed to faithfully represent the data and the stated criteria. For this reason, we study *temporal* OBQA.

In particular, we focus on *temporal conjunctive queries* (TCQs), which were originally proposed by [BBL13, BBL15c]. TCQs allow to combine conjunctive queries (CQs) via the Boolean operators and the temporal operators of propositional linear temporal logic LTL [Pnu77]. For example, the above criteria can be specified with the following TCQ $\phi(x)$, to obtain all eligible patients x:

$$\begin{aligned}&\Big(\Diamond^-\big(\exists y.\mathsf{HasFinding}(x,y)\wedge\mathsf{VZVInfection}(y)\big)\vee\\&\ \Diamond^-\big(\exists y.\mathsf{VaccinatedWith}(x,y)\wedge\mathsf{VZVVaccine}(y)\big)\Big)\\&\wedge\neg\big(\exists y.\mathsf{AllergyTo}(x,y)\wedge\mathsf{VZVVaccine}(y)\big)\end{aligned}$$

We here use the temporal operator 'some time in the past' ($\Diamond^-$) and consider the symbols $\mathsf{AllergyTo}$ and $\mathsf{VZVVaccine}$ to be *rigid*, which means that their interpretation does not change over time; e.g., we thus assume someone having an allergy to VZV vaccine to have this allergy for his whole life.

The semantics of TCQs is based on *temporal knowledge bases* (TKBs), which, in addition to the domain ontology (which is assumed to hold *globally*, i.e., at every point in time), contain finite *sequences* of fact bases. These fact bases represent the data associated to specific points in time—from the past until the *current time point n* ('now'). The problem we focus on is the evaluation of a TCQ w.r.t. such a temporal knowledge base, at the current time point.

In our setting, the information within the ontology and the fact bases does not explicitly refer to the temporal dimension, but is written in a *classical* (atemporal) description logic (DL); only the query is temporalized. In contrast, so-called *temporal DLs* [LWZ08, AKL$^+$07, AKRZ14, AKK$^+$14, GJS14, GBJS15, ABM$^+$14] extend classical DLs by temporal operators, which then occur within the ontology. However, as it is shown in [LWZ08, AKL$^+$07, AKRZ14, GJS14], most of these logics yield high reasoning complexities, even if the underlying atemporal DL allows for tractable reasoning. For that reason, lower complexities are only obtained by either considerably restricting the set of temporal operators or the DL.

A less expressive variant of TCQs called $\mathcal{ALC}$-LTL, which combines $\mathcal{ALC}$ axioms via LTL operators, has been introduced in [BGL12]. In [BBL13, BBL15c], the problem of answering TCQs over ontologies in the rather expressive DLs $\mathcal{ALC}$ and $\mathcal{SHQ}$ has been investigated (albeit without allowing transitive roles in the queries). However, reasoning in these DLs is not tractable anymore, and applications often need to process large quantities of data fast. Several lightweight logics, including *DL-Lite*, have been considered in [BLT15], but without negation in the TCQs; in contrast, we allow negation to occur in the queries as well as in the ontology language (*DL-Lite*$_{krom}$/*DL-Lite*$_{bool}$). [AKL$^+$07] also consider temporal variants of *DL-Lite*, but use less expressive formulas, similar to those

	Data Complexity			Combined Complexity		
	(i)	(ii)	(iii)	(i)	(ii)	(iii)
$DL\text{-}Lite^{[\,|\mathcal{H}]}_{[core|horn]}$	ALogTime	ALogTime	ALogTime	PSpace	PSpace	PSpace
$\mathcal{EL}$ [BT15c]	P	co-NP	co-NP	PSpace	PSpace	co-NExpTime
$\mathcal{ALC}$-$\mathcal{SHQ}$ [BBL15c]	co-NP	co-NP	≤ExpTime	ExpTime	co-NExpTime	2-ExpTime
$DL\text{-}Lite_{[krom|bool]}$	co-NP	co-NP	≤ExpTime	ExpTime	co-NExpTime	2-ExpTime
$DL\text{-}Lite^{\mathcal{H}}_{[krom|bool]}$	co-NP	co-NP	≤ExpTime	2-ExpTime	2-ExpTime	2-ExpTime

Table 1.1: Our results on the complexity of TCQ entailment compared to related work. All complexities except those marked with ≤ are tight.

of $\mathcal{ALC}$-LTL. In [BT15c], TCQs are studied in the context of the lightweight DL $\mathcal{EL}$, but it is shown that reasoning is quite hard if rigid symbols are considered. This motivates our study of TCQs in DLs of the *DL-Lite* family, which was tailored for (atemporal) query answering and allows for very efficient reasoning [CDL+07, GHJR+15]. Of particular interest in this setting is the question if temporal queries can be rewritten into first-order queries over a database, which can be expressed, e.g., as SQL queries, and executed using standard database systems; as it is possible in the atemporal case.

In this paper, we investigate the complexity of the TCQ entailment problem over temporal knowledge bases in several members of the *DL-Lite* family. In order of expressivity, we look at $DL\text{-}Lite_{core}$/$DL\text{-}Lite_{horn}$, their variants allowing role inclusions, and their counterparts $DL\text{-}Lite_{krom}$/$DL\text{-}Lite_{bool}$ featuring disjunctions on the right-hand side of concept inclusions, which can be used to express negated concepts. We regard both combined and data complexity and, as usual, distinguish three different settings regarding the rigid symbols:[3] (i) no symbols are allowed to be rigid, (ii) only rigid concept names are allowed, and (iii) both concepts and roles can be rigid.

Table 1.1 summarizes our results and shows that they are ambivalent. On the one hand, for expressive members of the *DL-Lite* family, we obtain at least the same complexities as for $\mathcal{SHQ}$. For logics below $DL\text{-}Lite^{\mathcal{H}}_{horn}$, however, we have results that are considerably better than those for $\mathcal{EL}$; above all, rigid roles can often be added without affecting the complexity. Unfortunately, our ALogTime lower bound for the data complexity of TCQ entailment in $DL\text{-}Lite_{core}$ shows that it is not possible to find a (pure) first-order rewriting of TCQs, in this setting; note that the graph of the parity function is in ALogTime and parity is not first-order definable [AB09]. The PSpace and co-NP lower bounds directly follow from the complexity of satisfiability in propositional LTL [SC85] and CQ entailment in $DL\text{-}Lite_{krom}$ [CDGL+05], respectively.

[3]Note that rigid concepts can be simulated by rigid roles [BGL12], even in $DL\text{-}Lite_{core}$.

2 Preliminaries

We first introduce several description logics of the *DL-Lite* family and then define TCQs over temporal ontologies formulated in these logics, as it was done for $\mathcal{ALC}$ in [BBL15c].

2.1 *DL-Lite* Description Logics

The various description logics of the so-called *DL-Lite* family extend the base formalism *DL-Lite*$_{core}$ by different concept constructors and/or kinds of expressions. We focus on several of the logics presented in [ACKZ09], which consider (different subsets of) the Boolean operators as concept constructors and so-called *role hierarchies*, abbreviated by the letter $\mathcal{H}$. We begin by recalling the syntax in the next definition.

Definition 2.1 (Syntax of *DL-Lite* Logics)**.** *Let* $\mathsf{N_C}$*,* $\mathsf{N_R}$*, and* $\mathsf{N_I}$*, respectively, be non-empty, pairwise disjoint sets of* concept names*,* role names*, and* individual names*. In the DL-Lite logics,* (basic) roles *R and* basic concepts *B are built from role names $P \in \mathsf{N_R}$ and concept names $A \in \mathsf{N_C}$ according to the following syntax rules:*

$$R ::= P \mid P^- \qquad\qquad B ::= A \mid \exists R$$

where $\cdot^-$ denotes the inverse role operator. $\mathsf{N_R^-}$ denotes the set of all roles. We consider the following axioms*: a* concept inclusion (CI) *is of the form*

$$B_1 \sqcap \ldots \sqcap B_m \sqsubseteq B_{m+1} \sqcup \ldots \sqcup B_{m+n}, \qquad (*)$$

where $B_1, \ldots, B_{m+n}$ are basic concepts; a role inclusion (RI) *is of the form*

$$R_1 \sqsubseteq R_2,$$

where $R_1, R_2 \in \mathsf{N_R^-}$; and an assertion *is of the form*

$$B(a) \text{ or } P(a,b),$$

where B is a basic concept, $P \in \mathsf{N_R}$, and $a, b \in \mathsf{N_I}$.

For $c \in \{core, horn, krom, bool\}$, we denote by DL-Lite$_c$ the logic that restricts concept inclusions of the form $(*)$ as follows:

- *m, n are arbitrary if $c = bool$;*
- *$m + n \leq 2$ if $c = krom$;*
- *$n \leq 1$ if $c = horn$; and*

- *$m + n \leq 2$ and $n \leq 1$ if $c = core$.*

If role inclusions are allowed in addition, this is indicated by a superscript $\mathcal{H}$, and we obtain the four DLs denoted by DL-Lite$_c^{\mathcal{H}}$.

Regarding a specific DL $\mathcal{L}$, an ontology *written in $\mathcal{L}$ is a finite set of concept and (if allowed in $\mathcal{L}$) role inclusions; and an* ABox *is a finite set of assertions. Together, an ontology $\mathcal{O}$ and an ABox $\mathcal{A}$, where the latter may contain only concept and role names that also occur in $\mathcal{O}$, form a* knowledge base (KB) $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$.

In our constructions, we also sometimes consider *negated* assertions of the form $\neg B(a)$ or $\neg P(a, b)$. As usual, the empty conjunction ($\sqcap$) is denoted by $\bot$ and the empty disjunction ($\sqcup$) by $\top$. We may write $B_1 \sqcap \cdots \sqcap B_m \sqsubseteq \neg B$ as abbreviation for $B_1 \sqcap \cdots \sqcap B_m \sqcap B \sqsubseteq \bot$, and $\bigsqcap \mathcal{B}$ for the conjunction $B_1 \sqcap \cdots \sqcap B_m$ if $\mathcal{B} = \{B_1, \ldots, B_m\}$. We further use the abbreviations $P^-(a, b) := P(b, a)$ and $(P^-)^- := P$, for $P \in \mathsf{N_R}$ and $a, b \in \mathsf{N_I}$.

Furthermore, we denote by $\mathsf{N_I}(\mathcal{K})$ the set of individual names that occur in the knowledge base $\mathcal{K}$, and by $\mathsf{N_C}(\mathcal{O})$ ($\mathsf{N_R^-}(\mathcal{O})$) the set of concept names (roles) that occur in the ontology $\mathcal{O}$. We use the notation $\mathsf{BC}(\mathcal{O})$ for the set of all basic concepts that can be built from $\mathsf{N_C}(\mathcal{O})$ and $\mathsf{N_R^-}(\mathcal{O})$, and $\mathsf{BC}^\neg(\mathcal{O})$ for the set $\mathsf{BC}(\mathcal{O})$ extended by negation.

We define the semantics as usual, in a model-theoretic way.

Definition 2.2 (Semantics of *DL-Lite* Logics)**.** *An* interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ *consists of a non-empty set $\Delta^\mathcal{I}$ (called* domain*), and an* interpretation function *$\cdot^\mathcal{I}$ that assigns to every $A \in \mathsf{N_C}$ a set $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$, to every $P \in \mathsf{N_R}$ a binary relation $P^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$, and to every $a \in \mathsf{N_I}$ an element $a^\mathcal{I} \in \Delta^\mathcal{I}$.*

This function is extended to all roles and concepts as follows:

$$(P^-)^\mathcal{I} := \{(y, x) \mid (x, y) \in R^\mathcal{I}\}; \text{ and}$$
$$(\exists R)^\mathcal{I} := \{x \mid \text{there is an } y \in \Delta^\mathcal{I} \text{ such that } (x, y) \in R^\mathcal{I}\}.$$

As usual, $\bot$ is interpreted as $\emptyset$ and $\top$ by $\Delta^\mathcal{I}$. The interpretation $\mathcal{I}$ satisfies *(or is a* model *of)*

- *a CI $B_1 \sqcap \cdots \sqcap B_m \sqsubseteq B_{m+1} \sqcup \cdots \sqcup B_{m+n}$ if $B_1^\mathcal{I} \cap \cdots \cap B_m^\mathcal{I} \subseteq B_{m+1}^\mathcal{I} \cup \cdots \cup B_{m+n}^\mathcal{I}$;*
- *an RI $R_1 \sqsubseteq R_2$ if $R_1^\mathcal{I} \subseteq R_2^\mathcal{I}$;*
- *a (negated) assertion $(\neg)B(a)$ if $a^\mathcal{I} \in B^\mathcal{I}$ ($a^\mathcal{I} \notin B^\mathcal{I}$);*
- *a (negated) assertion $(\neg)R(a, b)$ if $(a^\mathcal{I}, b^\mathcal{I}) \in R^\mathcal{I}$ ($(a^\mathcal{I}, b^\mathcal{I}) \notin R^\mathcal{I}$);*
- *a knowledge base if it satisfies all axioms contained in it.*

We write $\mathcal{I} \models \alpha$ if $\mathcal{I}$ satisfies the axiom α, $\mathcal{I} \models \mathcal{O}$ if $\mathcal{I}$ satisfies all CIs and RIs in the ontology $\mathcal{O}$, $\mathcal{I} \models \mathcal{A}$ if $\mathcal{I}$ satisfies all assertions in the ABox $\mathcal{A}$, and $\mathcal{I} \models \mathcal{K}$ if $\mathcal{I}$ is a model of the knowledge base $\mathcal{K}$. Further, a knowledge base $\mathcal{K}$ is said to be consistent *if it has a model, and $\mathcal{K}$* entails *an axiom α (written $\mathcal{K} \models \alpha$) if all models of $\mathcal{K}$ also satisfy α.*

Throughout the report, we assume that all interpretations $\mathcal{I}$ satisfy the *unique name assumption* (UNA), i.e., for all $a, b \in \mathsf{N_I}$ with $a \neq b$, we have $a^{\mathcal{I}} \neq b^{\mathcal{I}}$.

2.2 Temporal Conjunctive Queries

This report focuses on a temporal query language proposed in [BBL13], but we consider here knowledge bases formulated in *DL-Lite* instead of $\mathcal{ALC}$. The queries are formulas of propositional LTL, where the propositions are replaced by CQs, and are then answered over temporal knowledge bases, according to a semantics that is suitably lifted from propositional worlds to interpretations.

In the following, we assume (as in [BGL12, BBL15c]) that a subset of the concept and role names is designated as being *rigid* (as opposed to *flexible*). The intuition is that the interpretation of the rigid names is not allowed to change over time. In particular, the individual names are implicitly assumed to be rigid (i.e., an individual always has the same name). We denote by $\mathsf{N_{RC}} \subseteq \mathsf{N_C}$ the rigid concept names, and by $\mathsf{N_{RR}} \subseteq \mathsf{N_R}$ the rigid role names.

Definition 2.3 (Temporal Knowledge Base)**.** *A* temporal knowledge base (TKB) *$\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0 \leq i \leq n} \rangle$ consists of an ontology $\mathcal{O}$ and a finite sequence of ABoxes $\mathcal{A}_i$, where the latter only contain concept and role names that also occur in $\mathcal{O}$.*

Let $\mathfrak{I} = (\mathcal{I}_i)_{i \geq 0}$ be an infinite sequence of interpretations $\mathcal{I}_i = (\Delta, \cdot^{\mathcal{I}_i})$ over a non-empty domain Δ that is fixed (constant domain assumption). Then $\mathfrak{I}$ is a model of $\mathcal{K}$ (written $\mathfrak{I} \models \mathcal{K}$) if

- *for all $i \geq 0$, we have $\mathcal{I}_i \models \mathcal{O}$;*
- *for all i, $0 \leq i \leq n$, we have $\mathcal{I}_i \models \mathcal{A}_i$; and*
- *$\mathfrak{I}$ respects rigid names, i.e., $s^{\mathcal{I}_i} = s^{\mathcal{I}_j}$ for all symbols $s \in \mathsf{N_I} \cup \mathsf{N_{RC}} \cup \mathsf{N_{RR}}$ and $i, j \geq 0$.*

We use the notation $\mathsf{N_{RC}}(\mathcal{O})$, for the set of all rigid concept names that occur in $\mathcal{O}$, $\mathsf{BC}^{\neg}_{\mathsf{R}}(\mathcal{O})$ for the restriction of $\mathsf{BC}^{\neg}(\mathcal{O})$ to rigid concepts, and likewise for $\mathsf{BC_R}(\mathcal{O})$. We further denote by $\mathsf{N_I}(\mathcal{K})$ the set of all individual names occurring in the TKB $\mathcal{K}$.

As mentioned above, our query language combines conjunctive queries via LTL operators.

Definition 2.4 (Syntax of TCQs)**.** *Let* $\mathsf{N_V}$ *be a set of variables. A* conjunctive query *(CQ) is of the form* $\phi = \exists x_1, \ldots, x_m.\psi$*, where* $x_1, \ldots, x_m \in \mathsf{N_V}$ *and* ψ *is a (possibly empty) finite conjunction of atoms of the form*

- $A(t)$ *(*concept atom*), for* $A \in \mathsf{N_C}$ *and* $t \in \mathsf{N_I} \cup \mathsf{N_V}$*, or*
- $R(t_1, t_2)$ *(*role atom*), for* $R \in \mathsf{N_R}$ *and* $t_1, t_2 \in \mathsf{N_I} \cup \mathsf{N_V}$*.*

The empty conjunction is denoted by true*, and we write* $\alpha \in \phi$ *if the atom* α *occurs in* ϕ*.* Temporal conjunctive queries *(TCQs) are built from CQs as follows:*

- *each CQ is a TCQ; and*
- *if* ϕ_1 *and* ϕ_2 *are TCQs, then the following are also TCQs:*
  - $\neg\phi_1$ *(negation),* $\phi_1 \wedge \phi_2$ *(conjunction),*
  - $\bigcirc\phi_1$ *(next),* $\bigcirc^-\phi_1$ *(previous),*
  - $\phi_1 \mathbin{\mathsf{U}} \phi_2$ *(until), and* $\phi_1 \mathbin{\mathsf{S}} \phi_2$ *(since).*

We denote the set of individuals occurring in a TCQ ϕ by $\mathsf{N_I}(\phi)$, the set of variables occurring in ϕ by $\mathsf{N_V}(\phi)$, the set of free variables of ϕ by $\mathsf{N_{FV}}(\phi)$. A TCQ ϕ with $\mathsf{N_{FV}}(\phi) = \emptyset$ is called a *Boolean* TCQ. A *CQ-literal* is either a CQ or a negated CQ, and a *union of CQs* (UCQ) is a disjunction of CQs. As for role assertions, we may also use an expression of the form $R^-(t_1, t_2)$ to denote the role atom $R(t_2, t_1)$.

As usual, we use the following abbreviations: false, for $\neg\mathsf{true}$, $\phi_1 \vee \phi_2$ (disjunction), for $\neg(\neg\phi_1 \wedge \phi_2)$, $\Diamond\phi_1$ (eventually) for $\mathsf{true} \mathbin{\mathsf{U}} \phi_1$, $\Box\phi_1$ (always) for $\neg\Diamond\neg\phi_1$, and analogously for the past: $\Diamond^-\phi_1$ for $\mathsf{true} \mathbin{\mathsf{S}} \phi_1$, and $\Box^-\phi_1$ for $\neg\Diamond^-\neg\phi_1$.

We start by defining the semantics of CQs and TCQs for Boolean queries. As usual, it is given through the notion of homomorphisms [CM77].

Definition 2.5 (Semantics of TCQs)**.** *Let* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ *be an interpretation and* ψ *be a Boolean CQ. A mapping* $\pi \colon \mathsf{N_V}(\psi) \cup \mathsf{N_I}(\psi) \to \Delta^\mathcal{I}$ *is a homomorphism of* ψ *into* $\mathcal{I}$ *if*

- $\pi(a) = a^\mathcal{I}$*, for all* $a \in \mathsf{N_I}(\psi)$*;*
- $\pi(t) \in A^\mathcal{I}$*, for all concept atoms* $A(t)$ *in* ψ*; and*
- $(\pi(t_1), \pi(t_2)) \in R^\mathcal{I}$*, for all role atoms* $R(t_1, t_2)$ *in* ψ*.*

We say that $\mathcal{I}$ *is a* model *of* ψ *(written* $\mathcal{I} \models \psi$*) if there is such a homomorphism.*

Let now ϕ be a Boolean TCQ and $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ be an infinite sequence of interpretations. We define the satisfaction relation $\mathfrak{I}, i \models \phi$, where $i \geq 0$, by induction on the structure of ϕ:

$\mathfrak{I}, i \models \exists x_1, \dots, x_m.\psi$	*iff*	$\mathcal{I}_i \models \exists x_1, \dots, x_m.\psi$
$\mathfrak{I}, i \models \neg\phi_1$	*iff*	$\mathfrak{I}, i \not\models \phi_1$
$\mathfrak{I}, i \models \phi_1 \wedge \phi_2$	*iff*	$\mathfrak{I}, i \models \phi_1$ *and* $\mathfrak{I}, i \models \phi_2$
$\mathfrak{I}, i \models \bigcirc\phi_1$	*iff*	$\mathfrak{I}, i+1 \models \phi_1$
$\mathfrak{I}, i \models \bigcirc^{-}\phi_1$	*iff*	$i > 0$ *and* $\mathfrak{I}, i-1 \models \phi_1$
$\mathfrak{I}, i \models \phi_1 \mathsf{U} \phi_2$	*iff*	*there is some* $k \geq i$ *such that* $\mathfrak{I}, k \models \phi_2$ *and* $\mathfrak{I}, j \models \phi_1$, *for all* j, $i \leq j < k$
$\mathfrak{I}, i \models \phi_1 \mathsf{S} \phi_2$	*iff*	*there is some* k, $0 \leq k \leq i$, *such that* $\mathfrak{I}, k \models \phi_2$ *and* $\mathfrak{I}, j \models \phi_1$, *for all* j, $k < j \leq i$.

Given a TKB $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0\leq i\leq n}\rangle$, $\mathfrak{I}$ is called a model *of ϕ w.r.t. $\mathcal{K}$ if $\mathfrak{I} \models \mathcal{K}$ and $\mathfrak{I}, n \models \phi$. We call ϕ* satisfiable *w.r.t. $\mathcal{K}$ if it has a model w.r.t. $\mathcal{K}$. Furthermore, ϕ is* entailed *by $\mathcal{K}$ (written $\mathcal{K} \models \phi$) if every model of $\mathcal{K}$ is also a model of ϕ.*

Especially note that, as mentioned in the introduction, models of TCQs satisfy them at the current time point n.

We will often deal with conjunctions of CQ-literals ϕ. Since ϕ contains no temporal operators, the satisfaction of ϕ by an infinite sequence of interpretations $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ at time point i only depends on the interpretation $\mathcal{I}_i$. For simplicity, we then often write $\mathcal{I}_i \models \phi$ instead of $\mathfrak{I}, i \models \phi$. For the same reason, we use this notation also for UCQs. In this context, it is sufficient to deal with classical knowledge bases $\mathcal{K} = \langle \mathcal{O}, \mathcal{A}\rangle$, which can be seen as TKBs with only one ABox.

We now define the semantics of non-Boolean TCQs.

Definition 2.6 (Certain Answer)**.** *Let ϕ be a TCQ and $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0\leq i\leq n}\rangle$, be a TKB. The mapping $\mathfrak{a}\colon \mathsf{N_{FV}}(\phi) \to \mathsf{N_I}(\mathcal{K})$ is a* certain answer *to ϕ w.r.t. $\mathcal{K}$ if $\mathcal{K} \models \mathfrak{a}(\phi)$, where $\mathfrak{a}(\phi)$ denotes the Boolean TCQ that is obtained from ϕ by replacing the free variables according to $\mathfrak{a}$.*

As usual, the problem of computing all certain answers to a TCQ reduces to exponentially many entailment problems. We also assume that TCQs use only individual names that occur in the ABoxes, and only concept and role names that occur in the ontology; this is clearly without loss of generality.

Most of our upper bounds are based on the approach described in [BGL12, BBL15c]. We now introduce definitions that are important in this construction.

The *propositional abstraction* ϕ^{p} of a TCQ ϕ is built by replacing each CQ occurring in ϕ by a propositional variable such that there is a 1–1 relationship between the CQs $\alpha_1, \dots, \alpha_m$ occurring in ϕ and the propositional variables $p_1, \dots, p_m$ occurring in ϕ^{p}. The formula ϕ^{p} obtained in this way is a propositional LTL-formula [Pnu77].

Definition 2.7 (LTL). *Let $\{p_1, \dots, p_m\}$ be a finite set of propositional variables. An* LTL-formula ϕ *is built inductively from these variables using the constructors negation ($\neg\phi_1$), conjunction ($\phi_1 \wedge \phi_2$), next ($\bigcirc\phi_1$), previous ($\bigcirc^-\phi_1$), until ($\phi_1 \mathsf{U} \phi_2$), and since ($\phi_1 \mathsf{S} \phi_2$). An* LTL-structure *is an infinite sequence $\mathfrak{J} = (w_i)_{i \geq 0}$ of* worlds $w_i \subseteq \{p_1, \dots, p_m\}$. *The propositional variable p_j is satisfied by $\mathfrak{J}$ at $i \geq 0$ (written $\mathfrak{J}, i \models p_j$) if $p_j \in w_i$. The satisfaction of a complex propositional LTL-formula by an LTL-structure is defined as in Definition 2.5.*

For an LTL-formula ϕ, we use $\mathsf{Sub}(\phi)$ to denote the set of subformulas of ϕ. Note that the above definition extends the usual definition of LTL, which only considers the temporal operators $\bigcirc$ and U [Pnu77]. For this reason, this extended logic is often referred to as Past-LTL. An important result for this logic, the so-called separation theorem [Gab87], is given in the following proposition.

Proposition 2.8 ([Gab87]). *Every LTL-formula ϕ is equivalent to an LTL-formula in which no future operators occur in the scope of past operators and vice versa.*

Note that [Gab87] actually considers a slightly different temporal logic, using strict interpretations of S and U, and no other temporal operators. However, it is well-known that then $\bigcirc^-$ and $\bigcirc$ can be simulated. Conversely, it is easy to show that the strict versions of S and U can be expressed in our setting. Thus, the above result holds also for the temporal operators we consider here. Note that the size of the resulting "separated" LTL-formula may be non-elementary in the size of the original formula (i.e., the number of stacked exponents is determined by the number of alternations between past operators and future operators).

We call a propositional LTL formula a *future* formula if it contains no past operators and a *past* formula if it contains no future operators. Given a propositional LTL formula f, separated according to Proposition 2.8, we call a subformula g of f a *top-level* past (future) formula if it is of the form $\bigcirc^- g_1$, $\neg(\bigcirc^- g_1)$, $g_1 \mathsf{S} g_2$, or $\neg(g_1 \mathsf{S} g_2)$ ($\bigcirc g_1$, $\neg(\bigcirc g_1)$, $g_1 \mathsf{U} g_2$, or $\neg(g_1 \mathsf{U} g_2)$) and occurs in f at least once in the scope of no other temporal operator.

2.3 On Complexity

In this report, we study the complexity of TCQ entailment via the satisfiability problem, which has the same complexity as the complement of the entailment problem [BBL15c]. We consider two kinds of complexity measures: combined complexity and data complexity. For the combined complexity, all parts of the input, meaning the TCQ ϕ and the entire temporal knowledge base $\mathcal{K}$, are taken into account. In contrast, for the data complexity, the TCQ ϕ and the ontology $\mathcal{O}$ are assumed to be constant, and thus the complexity is measured only w.r.t. the data, the sequence of ABoxes.

Logic	Satisfiability Combined Complexity [ACKZ09]	UCQ Answering Combined Complexity [BAC10, BMP13, BMP14]	UCQ Answering Data Complexity [ACKZ09]
$DL\text{-}Lite_{core}^{[\ \|\mathcal{H}]}$	NLogSpace	NP	in AC^0
$DL\text{-}Lite_{horn}^{[\ \|\mathcal{H}]}$	P	NP	in AC^0
$DL\text{-}Lite_{krom}$	NLogSpace	in ExpTime	co-NP
$DL\text{-}Lite_{krom}^{\mathcal{H}}$	NLogSpace	?	co-NP
$DL\text{-}Lite_{bool}$	NP	ExpTime-hard	co-NP
$DL\text{-}Lite_{bool}^{\mathcal{H}}$	NP	2-ExpTime	co-NP

Table 2.9: Known results for the atemporal setting

Table 2.9 summarizes known complexity results for atemporal problems in the *DL-Lite* family, which are important for our work. We consider some complexity classes from the world of circuits:

$$AC^0 \subseteq NC^1 \subseteq AC^1,$$

which relate to the machine classes

$$\text{LogTime} \subseteq \text{ALogTime} \subseteq \text{NLogTime} \subseteq \text{LogSpace} \subseteq \text{NLogSpace} \subseteq \text{P}$$

such that $AC^0 \subseteq$ LogTime, ALogTime = DLogTime-uniform NC^1, and NLogSpace $\subseteq AC^1 \subseteq$ P. Note that the class AC^0 is of special interest for query answering in *DL-Lite*. This is because problems whose data complexity is in AC^0 can be solved by encoding them as first order (FO) queries over finite structures. Such problems are therefore also called *first order rewritable*.

Recall that we assumed all concept and role names in the ABox to also occur in the ontology. If this was not the case, we could simply add trivial axioms like $A \sqsubseteq A$ or $\exists R \sqsubseteq \exists R$ to $\mathcal{O}$ in order to satisfy this requirement. Although this formally increases the size of $\mathcal{O}$, these axioms do not affect the semantics of $\mathcal{O}$, and can thus be ignored in all reasoning problems involving $\mathcal{O}$. Hence, complexity results without this assumption remain valid in our setting.

3 Atemporal Canonical Models and Conjunctive Queries Revisited

In this section, we recall and extend known definitions and results, which we use in our proofs later in the report.

3.1 Canonical Models for Horn CIs

We consider $\textit{DL-Lite}^{\mathcal{H}}_{horn}$ and subsets of this logic and specify the notion of *canonical interpretation* for a knowledge base. This interpretation can be used for deciding consistency of the knowledge base and for answering CQs, because it contains those (prototypical) elements whose presence is enforced by the knowledge base. Then, it suffices to check whether the canonical interpretation is a model of a given knowledge base and if it satisfies a CQ, respectively. We use a construction based on the so-called *chase* [AHV95], similar to that proposed in [CDL+07] and [BAC10]; the latter extend the original definition of [CDL+07] to the logic $\textit{DL-Lite}^{\mathcal{H}}_{horn}$, and we further extend it. In particular, our canonical interpretation contains (unnamed) prototypical R-successors, $R \in \mathsf{N}_\mathsf{R}^-$, for all elements the knowledge base requires to satisfy $\exists R$; in contrast, [CDL+07, BAC10] only consider such prototypical successors if the knowledge base (i.e., the corresponding ABox) does not already identify a named individual to be such a successor. Unlike us, [CDL+07, BAC10] do also not consider arbitrary basic concept assertions, but only concept names.

We use the notation $c_{aR_1\dots R_\ell}$, for $a \in \mathsf{N_I}$ and $R_1, \dots, R_\ell \in \mathsf{N}_\mathsf{R}^-$, which is a domain element that acts as a prototypical R_ℓ-successor of a, if $\ell = 1$, and of $c_{aR_1\dots R_{\ell-1}}$, otherwise. For simplicity, we below assume that if $R_1 \sqsubseteq R_2$ is contained in an ontology $\mathcal{O}$, then we also have $\exists R_1 \sqsubseteq \exists R_2 \in \mathcal{O}$ and $\exists R_1^- \sqsubseteq \exists R_2^- \in \mathcal{O}$; and that $\mathcal{O}$ contains all trivial axioms of the form $B \sqsubseteq B$ for $B \in \mathsf{BC}(\mathcal{O})$.

Definition 3.1 (Canonical interpretation)**.** *Let $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$ be a $\textit{DL-Lite}^{\mathcal{H}}_{horn}$-knowledge base. We start defining the following sets, for all $A \in \mathsf{N_C}$, $P \in \mathsf{N_R}$,*

$$\begin{aligned} A^0 :=& \{a \mid A(a) \in \mathcal{A}\} \textit{ and} \\ P^0 :=& \{(a, b) \mid P(a, b) \in \mathcal{A}\} \cup \\ & \{(a, c_{aP}) \mid \exists P(a) \in \mathcal{A}\} \cup \\ & \{(c_{aP^-}, a) \mid \exists P^-(a) \in \mathcal{A}\}. \end{aligned}$$

Further, we define corresponding sets, for all $i > 0$, by applying the below rules. We denote with $(e, e') \in (P^-)^i$ the fact that $(e', e) \in P^i$. Similarly, $e \in (\exists R)^i$ denotes that there is an e' such that $(e, e') \in R^i$.

- *If $R_1 \sqsubseteq R_2 \in \mathcal{O}$ and $(e, e') \in R_1^i$, then we add (e, e') to R_2^{i+1}.*
- *If $\bigsqcap \mathcal{B} \sqsubseteq B \in \mathcal{O}$ and $e \in (B')^i$, for all $B' \in \mathcal{B}$, then*
  - *we add e to B^{i+1} if $B \in \mathsf{N_C}$;*
    otherwise, we have $B = \exists R$, $R \in \mathsf{N}_\mathsf{R}^-$, and,
  - *if $e \in \mathsf{N_I}(\mathcal{K})$, we add (e, c_{eR}) to R^{i+1};*
  - *else if $e = c_\varrho$, we add $(e, c_{\varrho R})$ to R^{i+1}.*

We collect the newly introduced individuals of the form c_ϱ *in the set* $\Delta_{\mathsf{u}}^{\mathcal{I}_\mathcal{O}}$*, and define the* canonical interpretation $\mathcal{I}_\mathcal{K}$ *for* $\mathcal{K}$ *as follows, for all* $a \in \mathsf{N_I}(\mathcal{A})$*,* $A \in \mathsf{N_C}$*, and* $P \in \mathsf{N_R}$*:*

$$\begin{aligned} \Delta^{\mathcal{I}_\mathcal{O}} &:= \mathsf{N_I}(\mathcal{A}) \cup \Delta_{\mathsf{u}}^{\mathcal{I}_\mathcal{O}}, \\ a^{\mathcal{I}_\mathcal{O}} &:= a, \\ A^{\mathcal{I}_\mathcal{O}} &:= \bigcup_{i=0}^{\infty} A^i, \text{ and} \\ P^{\mathcal{I}_\mathcal{O}} &:= \bigcup_{i=0}^{\infty} P^i. \end{aligned}$$

Note that the above assumptions about additional axioms in $\mathcal{O}$ ensure that, whenever $a \in (\exists R)^i$, then a has an R-successor of the form c_{aR}.

The rules given in the above definition correspond to the three rules proposed in [BAC10]. Further, the two above mentioned differences, regarding basic concept assertions and the additional successor individuals we consider, do not have special effects on reasoning. This is why we below sometimes refer to the results of [BAC10] without providing detailed proofs.

If $\mathcal{K}$ is inconsistent, then it is obvious that $\mathcal{I}_\mathcal{K}$ cannot be a model of $\mathcal{K}$. The converse of this statement is a little harder to show.

The proof proposed by [CDL⁺07, BAC10] is three-fold. First, it is shown that $\mathcal{I}_\mathcal{K}$ is a model of all *positive inclusions* (PIs) in $\mathcal{O}$, which are CIs whose right-hand side is not $\bot$. All other CIs are called *negative inclusions*. In order to check satisfiability of *DL-Lite*$^{\mathcal{H}}_{horn}$-KBs, negative inclusions must be considered. That is, if a negative inclusion in the ontology is violated by assertions of the ABox, then the knowledge base is inconsistent and hence unsatisfiable. Furthermore, an interaction of positive and negative inclusions may cause inconsistency. For these reasons, all negative inclusions implied by the ontology have to be considered and the so-called *closure* of the negative inclusions contained in $\mathcal{O}$ is regarded. The second step then consists of showing that $\mathcal{K}$ is consistent iff the assertions of the ABox do not contradict this closure. Third and last, it is shown that the latter is the case iff $\mathcal{I}_\mathcal{K}$ is a model of $\mathcal{K}$. The following proposition is a direct consequence of the above observations.

Proposition 3.2 ([BAC10, Lemma 3, Thm. 4])**.** *Let* $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$ *be a consistent DL-Lite*$^{\mathcal{H}}_{horn}$*-knowledge base, possibly including negated assertions. Then* $\mathcal{I}_\mathcal{K} \models \mathcal{K}$*.* □

The next proposition describes which basic concepts the elements of $\Delta^{\mathcal{I}_\mathcal{K}}$ satisfy, in dependence of the ABox.

Proposition 3.3. *Let* $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$ *be a consistent DL-Lite*$^{\mathcal{H}}_{horn}$*-knowledge base,* $e \in \Delta^{\mathcal{I}_\mathcal{K}}$*,* i *be the minimal number for which there is a symbol* S *such that* e *occurs* *in* S^i*,*

$$\mathcal{B} := \{A \in \mathsf{N_C}(\mathcal{O}) \mid e \in A^i\} \cup \{\exists R \mid R \in \mathsf{N_R^-}(\mathcal{O}),\ (e, e') \in R^i\}$$

be the set of corresponding basic concepts, and $B \in \mathsf{BC}(\mathcal{O})$*. Then, we have* $e \in B^{\mathcal{I}_\mathcal{K}}$ *iff* $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq B$*.*

Proof. For ($\Leftarrow$), we know that $e \in (B')^{\mathcal{I}_\mathcal{K}}$ for all $B' \in \mathcal{B}$ due to the definition of $\mathcal{I}_\mathcal{K}$. Hence, Proposition 3.2 yields the claim.

For ($\Rightarrow$), let j be the minimal index for which $e \in B^j$, which means that $j \geq i$. We show the claim by induction on j. If $j = i$, then $B \in \mathcal{B}$, and hence $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq B$ trivially holds.

If $j > i$, assume that the claim holds for all B' with $e \in (B')^{j-1}$. We consider the rule application which caused e to be contained in B^j.

- If it was caused by $R_1 \sqsubseteq R_2 \in \mathcal{O}$, then $B = \exists R_2^{(-)}$ and $e \in (\exists R_1^{(-)})^{j-1}$. By the induction hypothesis, $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R_1^{(-)} \sqsubseteq \exists R_2^{(-)}$.
- If it was caused by a CI $\bigsqcap \mathcal{B}' \sqsubseteq B \in \mathcal{O}$, then we know that $e \in (B')^{j-1}$ for all $B' \in \mathcal{B}'$. By the induction hypothesis, $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \bigsqcap \mathcal{B}' \sqsubseteq B$. □

The next proposition describes the basic concepts the new domain elements in $\Delta_\mathsf{u}^{\mathcal{I}_\mathcal{K}}$ satisfy in a straightforward way and hence shows that an element of the form $c_{\varrho R} \in \Delta_\mathsf{u}^{\mathcal{I}_\mathcal{O}}$ can indeed serve as a prototypical R-successor. The proposition directly follows from Definition 3.1 and Proposition 3.3.

Proposition 3.4. *Let* $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$ *be a consistent DL-Lite*$^\mathcal{H}_{horn}$*-knowledge base. Then, for all elements* $c_{\varrho R} \in \Delta_\mathsf{u}^{\mathcal{I}_\mathcal{K}}$ *and all* $B \in \mathsf{BC}(\mathcal{O})$*, we have* $c_{\varrho R} \in B^{\mathcal{I}_\mathcal{K}}$ *iff* $\mathcal{O} \models \exists R^- \sqsubseteq B$*.*

We conclude the section referring to a result which is rather important for us since we focus on query answering.

Proposition 3.5 ([BAC10, Thm. 9])**.** *For every UCQ* ψ *and every consistent DL-Lite*$^\mathcal{H}_{horn}$*-knowledge base* $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$*, possibly including negated assertions, we have* $\mathcal{K} \models \psi$ *iff* $\mathcal{I}_\mathcal{K} \models \psi$*.*

3.2 Atemporal Queries

In addition to the introductory definitions we define properties that specify CQs further.

For simplicity, we assume that all Boolean CQs we encounter are *connected*, meaning that all variables and individual names are related via chains of roles [RG10].

Definition 3.6 (Connected)**.** *A Boolean CQ ψ is called* connected *if, for all $t, t' \in \mathsf{N_I}(\psi) \cup \mathsf{N_V}(\psi)$, there is a sequence $t_1, \ldots, t_\ell \in \mathsf{N_I}(\psi) \cup \mathsf{N_V}(\psi)$ such that $t = t_1$ and $t' = t_\ell$ and for all i,$1 \leq i \leq \ell$, there is a $r \in \mathsf{N_R}$ such that either $r(t_i, t_{i+1}) \in \psi$ or $r(t_{i+1}, t_i) \in \psi$. A collection of Boolean CQs $\psi_1, \ldots, \psi_m$ is a partition of ψ if the atoms occurring in $\psi_1, \ldots, \psi_m$ are exactly the atoms of ψ, the sets $\mathsf{N_I}(\psi_i) \cup \mathsf{N_V}(\psi_i)$, $1 \leq i \leq m$, are pairwise disjoint, and each ψ_i is connected.*

It follows from a result in [Tes01], that we can assume Boolean TCQs to only contain connected CQs, without loss of generality. Furthermore note that, if a Boolean TCQ ϕ contains a CQ ψ that is not connected, then we can replace ψ by the conjunction $\psi_1 \wedge \cdots \wedge \psi_\ell$, where $\psi_1, \ldots, \psi_\ell$ is a partition of ψ. This conjunction is of linear size in the size of ψ and the resulting TCQ has exactly the same models as ϕ since every homomorphism of ψ into an interpretation $\mathcal{I}$ can be uniquely represented by a collection of homomorphisms of $\psi_1, \ldots, \psi_\ell$ into $\mathcal{I}$.

We next specify what we consider as *tree-shaped* CQs. Because of the inverse roles, the graphs described by the atoms of our CQs are not directed. For that reason, we consider structures similar to the *tree witnesses* defined in [KLT⁺10].

Definition 3.7 (Tree-shaped)**.** *Let ψ be a CQ with $\mathsf{N_I}(\psi) = \emptyset$ and $x \in \mathsf{N_V}(\psi)$, and $\mathcal{O}$ an ontology. A* tree witness *for x in ψ (w.r.t. $\mathcal{O}$) is a function of the form $\mathsf{f}\colon \mathsf{N_V}(\psi) \to (\mathsf{N_R^-} \times 2^{\mathsf{N_R^-}})^*$ such that*

- $\mathsf{f}(x) = \epsilon$;
- *for all $\varrho \cdot (R, \mathcal{C}) \in \mathsf{range}(\mathsf{f})$ and $S \in \mathcal{C}$, we have $\mathcal{O} \models R \sqsubseteq S$; and*
- *for all $S(y, y') \in \psi$, we have either*
  - $\mathsf{f}(y') = \mathsf{f}(y) \cdot (R, \mathcal{C})$ *with $\mathcal{O} \models S' \sqsubseteq S$ for some $S' \in \mathcal{C}$; or*
  - $\mathsf{f}(y) = \mathsf{f}(y') \cdot (R, \mathcal{C})$ *with $\mathcal{O} \models S' \sqsubseteq S^-$ for some $S' \in \mathcal{C}$.*

If a tree witness exists, then we call ψ tree-shaped*.*

Given a tree-shaped CQ ψ and a tree witness f for t in ψ, we denote by $\mathsf{Con}(\psi, \mathsf{f})$ the set of all sets $\mathcal{B} \subseteq \mathsf{BC}(\mathcal{O})$ such that

- *for each $A(y) \in \psi$ with $\mathsf{f}(y) = \epsilon$, we have $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq A$;*
- *for each $(R, \mathcal{C}) \in \mathsf{range}(\mathsf{f})$, we have $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R$;*
- *for each $A(y) \in \psi$ with $\mathsf{f}(y) = \varrho \cdot (R, \mathcal{C})$, we have $\mathcal{O} \models \exists R^- \sqsubseteq A$; and*
- *for all $\varrho \cdot (R_1, \mathcal{C}_1) \cdot (R_2, \mathcal{C}_2) \in \mathsf{range}(\mathsf{f})$, we have $\mathcal{O} \models \exists R_1^- \sqsubseteq \exists R_2$.*

Although the last two conditions in the definition of $\mathsf{Con}(\psi, \mathsf{f})$ do not refer to $\mathcal{B}$, they are needed to ensure that $\mathcal{B}$ induces the whole query. Hence, the set $\mathsf{Con}(\psi, \mathsf{f})$ is empty if they are not fulfilled for any tree witness f.

4 On Upper Bounds

In this section, we describe a general approach to solve the satisfiability problem (and thus the entailment problem), which has been proposed in [BBL15c, BGL12]. We further extend this approach such that it suits the complexity results we want to show in the context of *DL-Lite*. This procedure is then used in later sections to obtain several upper bounds.

4.1 A General Approach for Solving Satisfiability

In a nutshell, the satisfiability problem of a TCQ w.r.t. a TKB is reduced to two separate satisfiability problems—one in LTL and one in *DL-Lite*. We describe this approach, in the following. To this end, let $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0 \leq i \leq n} \rangle$ be a TKB and ϕ be a Boolean TCQ. For the LTL part, we now consider the propositional abstraction ϕ^{p} of ϕ, which contains the propositional variables $p_1, \ldots, p_m$ in place of the CQs $\alpha_1, \ldots, \alpha_m$ occurring in ϕ. Let them be such that α_i was replaced by p_i, for $1 \leq i \leq m$. Furthermore, we consider a set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1, \ldots, p_m\}}$, which specifies the worlds that are allowed to occur in an LTL-structure satisfying ϕ^{p}, and a mapping $\iota \colon \{0, \ldots, n\} \to \{1, \ldots, k\}$ that fixes the worlds belonging to the first $n+1$ time points, which need to be consistent with the ABoxes.

Definition 4.1 (t-satisfiability). *The LTL-formula ϕ^{p} is* t-satisfiable w.r.t. $\mathcal{S}$ and ι *if there exists an LTL-structure $\mathfrak{J} = (w_i)_{i \geq 0}$ such that*

- $\mathfrak{J}, n \models \widehat{\phi}$,
- $w_i \in \mathcal{S}$ *for all* $i \geq 0$, *and*
- $w_i = X_{\iota(i)}$ *for all* i, $0 \leq i \leq n$.

However, it is not sufficient to guess $\mathcal{S}$ and ι and to then check the above condition. We must also ensure that $\mathcal{S}$ can indeed be induced by some sequence of interpretations that is a model of $\mathcal{K}$. The following definition introduces a condition that needs to be satisfied for this to hold. That is, it covers the part of satisfiability regarding *DL-Lite*.

Definition 4.2 (r-satisfiability). *The set $\mathcal{S}$ is called* r-satisfiable w.r.t. ι and $\mathcal{K}$ *if there are interpretations $\mathcal{J}_1, \ldots, \mathcal{J}_k$, $\mathcal{I}_0, \ldots, \mathcal{I}_n$ such that*

- *the interpretations share the same domain and respect rigid names*[4]*;*
- *the interpretations are models of* $\mathcal{O}$*;*

[4] This is defined analogously to the case of sequences of interpretations (cf. Definition 2.3).

- *for all i, $1 \leq i \leq k$, $\mathcal{J}_i$ is a model of $\chi_i := \bigwedge_{p_j \in X_i} \alpha_j \wedge \bigwedge_{p_j \in \overline{X_i}} \neg\alpha_j$, where $\overline{X_i} := \{p_1, \ldots, p_m\} \setminus X_i$; and*
- *for all i, $0 \leq i \leq n$, $\mathcal{I}_i$ is a model of $\mathcal{A}_i$ and $\chi_{\iota(i)}$.*

Note that, through the existence of the interpretations $\mathcal{J}_i$, $1 \leq i \leq k$, it is ensured that the conjunction χ_i of the CQ-literals specified by X_i is consistent. A set $\mathcal{S}$ containing a set X_i for which this does not hold cannot be induced by any sequence of interpretations that are models of $\mathcal{O}$. Moreover, the ABoxes are considered through the interpretations $\mathcal{I}_i$, $0 \leq i \leq n$, which represent the first $n + 1$ interpretations in such a sequence.

These two checks together suffice to determine the satisfiability of ϕ w.r.t. $\mathcal{K}$.

Lemma 4.3 ([BBL15c, Lemma 4.7])**.** *The TCQ ϕ has a model w.r.t. the TKB $\mathcal{K}$ iff there are a set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1,\ldots,p_m\}}$ and a mapping $\iota\colon \{0, \ldots, n\} \to \{1, \ldots, k\}$ such that*

- *ϕ^{p} is t-satisfiable w.r.t. $\mathcal{S}$ and ι, and*
- *$\mathcal{S}$ is r-satisfiable w.r.t. ι and $\mathcal{K}$.* □

The original proof in [BBL15c] considers only the DL $\mathcal{SHQ}$, but is actually independent of the logic under consideration, and hence also applies in our setting.

The remaining parts of this report focus on the question how the three problems of (i) obtaining $\mathcal{S}$ and ι, (ii) solving the LTL satisfiability test, and (iii) solving the r-satisfiability test(s), can be solved.

At this point, we do not follow the approach of [BBL15c] further, but propose another similar to that of [BT15c, BT15b], which later allows us to obtain the especially low complexity results for *DL-Lite*$^{\mathcal{H}}_{horn}$. In particular, the rest of this section refers to problem (iii). Further details regarding the solution of problems (i) and (ii) are given in the subsequent sections.

4.2 On Checking r-satisfiablility

In this section, we describe how to guess a polynomial amount of additional information such that the r-satisfiability test for $\mathcal{S}$ can be split into independent satisfiability tests for the individual time points (see Definition 4.8). To describe this, we need the notions of *consequences* and *witnesses*.

4.2.1 Consequences, Witnesses, and Witness Queries

Definition 4.4 (Consequences)**.** *For a CQ α, let α' denote the CQ obtained by instantiating all variables x in α with fresh individual names a_x. The set of* consequences *of* α *is defined as*

$$\mathcal{C}_{\mathcal{O}}(\alpha) := \{C(a) \mid C \in \mathsf{BC}_{\mathsf{R}}^{-}(\mathcal{O}),\ a \in \mathsf{N_I}(\alpha'),\ \mathcal{O} \models \bigsqcap \mathsf{BC}^{-}(a,\alpha') \sqsubseteq C\} \cup \\ \{R(a,b) \mid R \in \mathsf{N}_{\mathsf{RR}}^{-}(\mathcal{O}),\ S(a,b) \in \alpha',\ \mathcal{O} \models S \sqsubseteq R\},$$

where

$$\mathsf{BC}^{-}(a,\alpha') := \{A \in \mathsf{N_C} \mid A(a) \in \alpha'\} \cup \{\exists R \mid R \in \mathsf{N}_{\mathsf{R}}^{-},\ R(a,b) \in \alpha'\}.$$

We collect all the new individual names a_x in the set $\mathsf{N}_{\mathsf{I}}^{\mathsf{aux}}$.

The consequences of a CQ describe those structures that, if the CQ is satisfied at one time point, have to exist at all other time points, because of the agreement on the rigid names. However, by using such assertions, we cannot capture the shared domain, in the sense that we cannot enforce the structures to be instantiated by the same individuals, at all time points. Nevertheless, in the context of *DL-Lite*$^{\mathcal{H}}_{horn}$, we cannot express meaningful information regarding role successors, either. Hence, we later show that, if information from different time points is relevant at other time points, it suffices to know about the existence of such rigid structures. In particular, at any time point, we cannot enforce (e.g., by requiring certain CQs to be satisfied) the satisfaction of certain rigid concepts or roles in addition to a specific structure if we do not consider named individuals. Further note that the set of consequences may also contain negative assertions.

In the following, we introduce sets $\mathcal{B} = \{B_1, \dots, B_\ell\} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ that can be considered witnesses for the satisfaction of certain concepts or the existence of certain elements in a canonical model. In an abuse of notation, we may write $\mathcal{B}(x)$ for the conjunction $B_1(x) \wedge \dots \wedge B_\ell(x)$, and $\mathcal{B}(x) \subseteq \psi$ to express that the conjuncts of $\mathcal{B}(x)$ are part of the CQ ψ. Note that the definition of CQs (cf. Definition 2.4) does not allow basic concepts of the form $\exists R(x), R \in \mathsf{N}_{\mathsf{R}}^{-}$, to occur in CQs. However, we can obviously replace such atoms by atoms of the form $R(x,y)$ if $R \in \mathsf{N_R}$ and $R(y,x)$ otherwise, if we extend the set of existentially quantified variables of ψ with a fresh variable y, correspondingly.

Definition 4.5 (Witness)**.** *Let* $\mathcal{O}$ *be an ontology. A set* $\mathcal{B} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ *is a* witness *of a basic concept* $B \in \mathsf{BC}(\mathcal{O})$ *w.r.t.* $\mathcal{O}$ *if there are* $R_1, \dots, R_\ell \in \mathsf{N_R}$*,* $\ell \geq 1$*, such that* $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R_1$*,* $\mathcal{O} \models \exists R_i^- \sqsubseteq \exists R_{i+1}$*,* $1 \leq i \leq \ell - 1$*, and* $\mathcal{O} \models \exists R_\ell^- \sqsubseteq B$*.*

Let further $\mathcal{I}$ *be the canonical interpretation for* $\langle \mathcal{O}, \mathcal{A} \rangle$*, where* $\mathcal{A}$ *is an arbitrary ABox. Then,* $\mathcal{B}$ *is a* witness *of an element* $c_{\varrho R_0 \dots R_\ell} \in \Delta_{\mathsf{u}}^{\mathcal{I}}$ *w.r.t.* $\langle \mathcal{O}, \mathcal{A} \rangle$ *if* $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R_0$ *and* $c_\varrho \in (\bigsqcap \mathcal{B})^{\mathcal{I}}$ *or* $\varrho \in \mathsf{N_I}(\mathcal{A}) \cap (\bigsqcap \mathcal{B})^{\mathcal{I}}$*.*

The set of all witnesses of a basic concept or unnamed element α *w.r.t.* $\mathcal{O}$ *is denoted by* $\mathcal{W}_{\mathcal{O}}(\alpha)$*.*

Intuitively, the witnesses for concepts w.r.t. some $\mathcal{O}$ specify alternatives for rigid basic concepts, whose instantiation leads to the instantiation of the considered concept in models of $\mathcal{O}$. Furthermore, regarding some canonical interpretation $\mathcal{I}$, the witnesses for elements $c_\varrho \in \Delta^{\mathcal{I}}_{\mathsf{u}}$, describe rigid basic concepts whose instantiation by (not necessarily direct) role predecessors in $\mathcal{I}$ causes the existence of c_ϱ.

We now lift this notion to tree-shaped CQs, which can be witnessed by rigid CQs as follows.

Definition 4.6 (Tree witness query)**.** *Let* f *be a tree witness for a CQ* α *w.r.t.* $\mathcal{O}$ *and* $\mathcal{B} \in \mathsf{Con}(\alpha, \mathsf{f})$*. We denote by* $\mathcal{B}|_{\mathsf{R}}$ *the set* $\mathcal{B} \cap \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$*. Let further* ψ *be a CQ over the variables of the form* y_ϱ *for* $\varrho \in \mathsf{range}(\mathsf{f})$*.*

We call $\varrho \in \mathsf{range}(\mathsf{f})$ rigidly witnessed *in* ψ *(w.r.t.* $\mathcal{B}$ *and* f*) if*

- $\varrho = \varrho_1 \cdot (R, \mathcal{C})$ *and* ϱ_1 *is rigidly witnessed in* ψ*;*
- $\varrho = (R, \mathcal{C})$ *and there is a set* $\mathcal{B}_{\exists R} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ *such that* $\mathcal{O} \models \bigsqcap \mathcal{B}_{\exists R} \sqsubseteq \exists R$ *and* $\mathcal{B}_{\exists R}(y_\epsilon) \subseteq \psi$*; or*
- $\varrho = \varrho_1 \cdot (R_1, \mathcal{C}_1) \cdot (R_2, \mathcal{C}_2)$ *and there is a set* $\mathcal{B}_{\exists R_2} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ *such that* $\mathcal{O} \models \bigsqcap \mathcal{B}_{\exists R_2} \sqsubseteq \exists R_2$ *and* $\mathcal{B}_{\exists R_2}(y_{\varrho_1 \cdot (R_1, \mathcal{C}_1)}) \subseteq \psi$*.*

The CQ ψ *is a* tree witness query *for* α *(w.r.t.* $\mathcal{O}$*,* $\mathcal{B}$ *and* f*) if it is minimal (w.r.t. set inclusion regarding the set of atoms) among all CQs satisfying the following conditions:*

- $\mathcal{B}|_{\mathsf{R}}(y_\epsilon) \subseteq \psi$*;*
- *for each* $A(y) \in \alpha$ *with* $\mathsf{f}(y) = \epsilon$*, we have (i)* $A \in \mathsf{N_{RC}}$ *and* $A(y_\epsilon) \in \psi$ *or (ii)* $\mathcal{O} \models \bigsqcap \mathcal{B}|_{\mathsf{R}} \sqsubseteq A$*;*
- *for each* $A(y) \in \alpha$ *with* $\mathsf{f}(y) = \varrho \cdot (R, \mathcal{C})$*, we have (i)* $\mathsf{f}(y)$ *is rigidly witnessed in* ψ *or (ii) there is a set* $\mathcal{B}_A \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ *with* $\mathcal{O} \models \bigsqcap \mathcal{B}_A \sqsubseteq A$ *and* $\mathcal{B}_A(y_{\mathsf{f}(y)}) \subseteq \psi$*;*
- *for each* $\varrho \cdot (R, \mathcal{C}) \in \mathsf{range}(\mathsf{f})$*, we have (i)* $\varrho \cdot (R, \mathcal{C})$ *is rigidly witnessed in* ψ*, or (ii)* $\mathcal{C} \subseteq \mathsf{N_{RR}}$ *and* $S(y_\varrho, y_{\varrho \cdot (R, \mathcal{C})}) \in \psi$ *for all* $S \in \mathcal{C}$*.*

An important property of tree witness queries is that they contain only rigid concept and role names. We now slightly extend this notion to *witness queries*, while preserving the above property.

Definition 4.7 (Witness query)**.** *Let* $i \in \{0, \dots, n+k\}$*. A* witness query ψ *for a tree-shaped CQ* α *(w.r.t.* $\mathcal{O}$*) is a CQ such that either*

- ψ *is a tree witness query for* α *w.r.t.* $\mathcal{O}$*;*
- *there are* $R \in \mathsf{N}^-_{\mathsf{R}}(\mathcal{O})$*,* $\mathcal{B} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$*, and a tree witness* f *for* α *w.r.t.* $\mathcal{O}$ *such that* $\mathcal{B}$ *is a witness of* $\exists R$ *w.r.t.* $\mathcal{O}$*,* $\{\exists R\} \in \mathsf{Con}(\alpha, \mathsf{f})$*, and* $\psi = \exists x.\mathcal{B}(x)$*.*

4.2.2 R-Complete Tuples

Using the notions of consequences and witness queries, we can now describe our approach to solve the r-satisfiability problem.

Let now ϕ be a TCQ and $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0 \le i \le n} \rangle$ be a TKB formulated in $\mathit{DL\text{-}Lite}^{\mathcal{H}}_{horn}$. We further assume that a set $\mathcal{S} = \{X_1, \dots, X_k\} \subseteq 2^{\{p_1,\dots,p_m\}}$ and a mapping $\iota\colon \{0, \dots, n\} \to \{1, \dots, k\}$ are given. We later describe how to actually obtain $\mathcal{S}$ and ι to show our upper bounds.

We denote by $\mathcal{Q}_\phi$ the set of CQs occurring in ϕ, and assume without loss of generality that they use disjoint variables.[5] For ease of presentation, we denote by Q_i the set $\{\alpha_j \mid p_j \in X_i\}$, and by $\mathcal{A}_{Q_i}$ the ABox obtained from Q_i by instantiating all variables x in the CQs $\alpha_j \in Q_i$ with the corresponding individual names a_x from $\mathsf{N}_\mathsf{I}^{\mathsf{aux}}$. For ease of presentation, for all i, $1 \le i \le k$, we define the set $\mathcal{A}_{n+i} := \emptyset$ and extend ι such that $\iota(n+i) := i$.

In the following, we consider tuples of the form $(\mathcal{A}_\mathsf{R}, Q_\mathsf{R}, Q_\mathsf{R}^{\neg}, R_\mathsf{F})$, where

- $\mathcal{A}_\mathsf{R}$ is an *ABox type* for $\mathcal{K}$, that is, a set of rigid (negated) assertions $(\neg)\alpha$ formulated over $\mathsf{N}_\mathsf{I}(\mathcal{K})$, $\mathsf{BC}_\mathsf{R}(\mathcal{O})$, and $\mathsf{N}_\mathsf{RR}^{-}(\mathcal{O})$, with the property that $\alpha \in \mathcal{A}_\mathsf{R}$ iff $\neg\alpha \notin \mathcal{A}_\mathsf{R}$.
- $Q_\mathsf{R}, Q_\mathsf{R}^{\neg} \subseteq \mathcal{Q}_\phi$.
- $R_\mathsf{F} \subseteq \{\exists S(b) \mid S \in \mathsf{N}_\mathsf{R}(\mathcal{O}) \setminus \mathsf{N}_\mathsf{RR},\ b \in \mathsf{N}_\mathsf{I}(\mathcal{K})\}$.

Given such a tuple, we define the atemporal KBs

$$\mathcal{K}_\mathsf{R}^i := \langle \mathcal{O}, \mathcal{A}_\mathsf{R} \cup \mathcal{A}_{Q_\mathsf{R}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_{R_\mathsf{F}} \cup \mathcal{A}_i \rangle$$

for all i, $0 \le i \le n+k$, where $\mathcal{A}_{Q_\mathsf{R}} := \bigcup_{\alpha \in Q_\mathsf{R}} \mathcal{C}_\mathcal{O}(\alpha)$ and $\mathcal{A}_{R_\mathsf{F}} := \bigcup_{\exists S(b) \in R_\mathsf{F}} \mathcal{A}_{\exists S(b)}$ is defined as follows: for every domain element $c_{b\varrho}$ of the canonical model $\mathcal{I}_{\langle \mathcal{O}, \{\exists S(b)\}\rangle}$ such that $|\varrho| \le \max\{|\mathsf{N}_\mathsf{V}(\psi)| \mid \psi \in \mathcal{Q}_\phi\}$, we introduce a new individual name $a_{b\varrho}$ and add the following assertions to $\mathcal{A}_{\exists S(b)}$, for every $a_{b\varrho R}$ of this form:

- For every $B \in \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $\mathcal{O} \models \exists R^- \sqsubseteq B$, we add the concept assertion $B(a_{b\varrho R})$.
- For every $R' \in \mathsf{N}_\mathsf{RR}(\mathcal{O})$ with $\mathcal{O} \models R \sqsubseteq R'$, we add the role assertion $R'(a_{b\varrho}, a_{b\varrho R})$, where we set $a_b := b$.

We collect all the individual names $a_{b\varrho}$ into the set $\mathsf{N}_\mathsf{I}^{\mathsf{tree}}$. The purpose of the ABoxes $\mathcal{A}_{\exists S(b)}$ is to formalize all rigid consequences of a (flexible) S-successor of b, which have to be present at all time points. We need to consider these

[5] If this was not the case, we could simply rename them.

consequences only up to a depth that ensures that all matches of CQs depending on this flexible S-successor can be fully characterized.

This construction is needed in the case that CQs from $\mathcal{Q}_\phi$ contain role atoms involving roles that have rigid subroles, which in turn have flexible subroles. Unfortunately, the resulting ABox $\mathcal{A}_{R_\mathsf{F}}$ is of exponential size. We later show in Section 5 that this does not affect the combined complexity of PSPACE; however, for our data complexity results we have to disallow such roles to occur in queries. Formally, a role S is called *r-simple* (w.r.t. $\mathcal{O}$) if there are no roles R, R' such that R is flexible, R' is rigid, and $\mathcal{O} \models R \sqsubseteq R' \sqsubseteq S$. We extend this notion to queries by requiring that all role atoms in *r-simple* TCQs involve only r-simple roles. Note that if we disallow role hierarchies or rigid roles, then all roles are necessarily r-simple.

We can now formalize the properties that the tuple $(\mathcal{A}_\mathsf{R}, Q_\mathsf{R}, Q_\mathsf{R}^\neg, R_\mathsf{F})$ has to fulfill in order to characterize the r-satisfiability of $\mathcal{S}$ w.r.t. ι and $\mathcal{O}$.

Definition 4.8 (r-complete)**.** *A tuple* $(\mathcal{A}_\mathsf{R}, Q_\mathsf{R}, Q_\mathsf{R}^\neg, R_\mathsf{F})$ *as above is* r-complete *(w.r.t.* $\mathcal{S}$ *and* ι*) if the following hold:*

(R1) *For all* $i \in \{0, \dots, n+k\}$*,* $\mathcal{K}_\mathsf{R}^i$ *is consistent.*

(R2) *For all* $i \in \{0, \dots, n+k\}$ *and* $p_j \in \overline{X_{\iota(i)}}$*, we have* $\mathcal{K}_\mathsf{R}^i \not\models \alpha_j$*.*

(R3) *If there is an* $X \in \mathcal{S}$ *such that* $p_j \in X$*, then* $\alpha_j \in Q_\mathsf{R}$*.*

(R4) *If there is an* $X \in \mathcal{S}$ *such that* $p_j \in \overline{X}$*, then* $\alpha_j \in Q_\mathsf{R}^\neg$*.*

(R5) *For all* $i \in \{0, \dots, n+k\}$*, all tree-shaped CQs* $\alpha \in Q_\mathsf{R}^\neg$*, and all witness queries* ψ *for* α *w.r.t.* $\mathcal{O}$*, we have* $\mathcal{K}_\mathsf{R}^i \not\models \psi$*.*

(R6) *If* ϕ *is not r-simple, then for all* $S \in \mathsf{N_R}(\mathcal{O}) \setminus \mathsf{N_{RR}}$ *and* $b \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$*, we have* $\exists S(b) \in R_\mathsf{F}$ *iff there is an index* $i \in \{0, \dots, n+k\}$ *such that* $\langle \mathcal{O}, \mathcal{A}_\mathsf{R} \cup \mathcal{A}_{Q_\mathsf{R}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_i \rangle \models \exists S(b)$*.*

If ϕ is r-simple, then R_F does not need to satisfy any condition, and hence we can assume that it is empty, in which case $\mathcal{A}_{R_\mathsf{F}}$ is also empty. The ABoxes $\mathcal{A}_{Q_\mathsf{R}}$, $\mathcal{A}_{Q_x}$, and $\mathcal{A}_{R_\mathsf{F}}$ may share some individual names from $\mathsf{N_I^{aux}}$, whereas different ABoxes $\mathcal{A}_{Q_x}$ and $\mathcal{A}_{Q_y}$ do not share any individual names from $\mathsf{N_I^{aux}}$ since we assume the CQs to use disjoint variables.

The idea is to fix the interpretation of the rigid names on all named individuals ($\mathcal{A}_\mathsf{R}$) and specify two sets of CQs that, respectively, describe relations which always exist by specifying the CQs that are satisfied at least once (Q_R), and CQs that are allowed to occur negatively in $\mathcal{S}$ ($Q_\mathsf{R}^\neg$). In particular, note that Q_R and $Q_\mathsf{R}^\neg$ may overlap. The first two conditions ensure that, for all ABoxes $\mathcal{A}_i$, $0 \leq i \leq n$, and worlds X_{i-n}, $n+1 \leq i \leq n+k$, to be considered, we have that exactly the queries specified by $X_{\iota(i)}$ can be satisfied w.r.t. $\mathcal{O}$ (together with the assertions from $\mathcal{K}_{\mathsf{R}}^i$). Based on Q_{R}, the third condition ensures that rigid relations that need to exist in the interpretation induced by some world $X \in \mathcal{S}$ always exist. The fourth condition checks that only the queries from $Q_{\mathsf{R}}^{\neg}$ can occur negatively in any $X \in \mathcal{S}$. The fifth condition ensures that there is a model of $\mathcal{K}_{\mathsf{R}}^i$ that does not satisfy any of the witness queries of the CQs in $Q_{\mathsf{R}}^{\neg}$. Finally, the sixth condition ensures that only such rigid consequences of $\exists S(b)$ are included in $\mathcal{A}_{R_{\mathsf{F}}}$ that are implied already by one of the $\mathcal{K}_{\mathsf{R}}^i$ without $\mathcal{A}_{R_{\mathsf{F}}}$.

In the remainder of this section we show that the existence of an r-complete tuple w.r.t. $\mathcal{S}$ fully characterizes the r-satisfiability of $\mathcal{S}$.

Lemma 4.9. *$\mathcal{S}$ is r-satisfiable w.r.t. ι and $\mathcal{O}$ iff there is an r-complete tuple w.r.t. $\mathcal{S}$ and ι.*

The proof of this lemma is split over the following two subsections.

4.2.3 If $\mathcal{S}$ is r-satisfiable w.r.t. ι and $\mathcal{O}$, then there is an r-complete tuple w.r.t. $\mathcal{S}$ and ι.

We first show that the satisfaction of a witness query for a CQ $\alpha \in \mathcal{Q}_\phi$ implies the satisfaction of α.

Lemma 4.10. *Let $\alpha \in \mathcal{Q}_\phi$, ψ be a witness query for α, and $\mathcal{I}$ be a model of $\mathcal{O}$. Then, $\mathcal{I} \models \psi$ implies $\mathcal{I} \models \alpha$.*

Proof. Let π be the homomorphism of ψ into $\mathcal{I}$, and consider first the case that ψ is a tree witness query w.r.t. some tree witness f and $\mathcal{B} \in \mathsf{Con}(\alpha, \mathsf{f})$. We define a homomorphism π' of α into $\mathcal{I}$ using an auxiliary mapping $\pi'\colon \mathsf{range}(\mathsf{f}) \to \Delta^{\mathcal{I}}$, for which we then set $\pi'(y) := \pi'(\mathsf{f}(y))$ for all $y \in \mathsf{N_V}(\psi)$. We start by defining $\pi'(\varrho) := \pi(y_\varrho)$ for all $\varrho \in \mathsf{range}(\mathsf{f})$ that are not rigidly witnessed in ψ. Observe that if ϱ is rigidly witnessed in ψ, then the variable y_ϱ does not occur in ψ; furthermore, $\varrho = \epsilon$ is never rigidly witnessed and hence at least y_ϵ must occur in ψ.

We define π' on the remaining elements $\varrho \in \mathsf{range}(\mathsf{f})$ by induction on the structure of f. We first consider the case that ϱ is *directly* rigidly witnessed, i.e., no prefix of ϱ is rigidly witnessed.

- If $\varrho = (R, \mathcal{C})$, then there must be a $\mathcal{B}_{\exists R} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ with $\mathcal{O} \models \bigsqcap \mathcal{B}_{\exists R} \sqsubseteq \exists R$ and $\mathcal{B}_{\exists R}(y_\epsilon) \subseteq \psi$. Since π is a homomorphism of ψ into $\mathcal{I}$ and $\mathcal{I} \models \mathcal{O}$, we get $\pi'(\epsilon) = \pi(y_\epsilon) \in (\bigsqcap \mathcal{B}_{\exists R})^{\mathcal{I}} \subseteq (\exists R)^{\mathcal{I}}$. Hence, there must exist an element $e \in \Delta^{\mathcal{I}}$ with $(\pi'(\epsilon), e) \in R^{\mathcal{I}}$. We set $\pi'(R, \mathcal{C}) := e$.

- If $\varrho = \varrho_1 \cdot (R_1, \mathcal{C}_1) \cdot (R_2, \mathcal{C}_2)$, then there is a set $\mathcal{B}_{\exists R_2} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ such that $\mathcal{O} \models \bigsqcap \mathcal{B}_{\exists R_2} \sqsubseteq \exists R_2$ and $\mathcal{B}_{\exists R_2}(y_{\varrho_1 \cdot (R_1, \mathcal{C}_1)}) \subseteq \psi$. As above, we can define $\pi'(\varrho)$ to be the R-successor of $\pi'(\varrho_1 \cdot (R_1, \mathcal{C}_1))$ that must exist in $\mathcal{I}$ because of the above conditions.

For the induction step, let $\varrho = \varrho_1 \cdot (R_1, \mathcal{C}_1) \cdot (R_2, \mathcal{C}_2)$ be such that $\varrho_1 \cdot (R_1, \mathcal{C}_1)$ is rigidly witnessed and $\pi'(\varrho \cdot (R_1, \mathcal{C}_1))$ has already been defined. By the construction above, we know that $\pi'(\varrho \cdot (R_1, \mathcal{C}_1)) \in (\exists R_1^-)^\mathcal{I}$. By the last condition of $\mathsf{Con}(\alpha, \mathsf{f})$, there must be an R_2-successor of $\pi'(\varrho \cdot (R_1, \mathcal{C}_1))$ in $\mathcal{I}$, which we can choose as $\pi'(\varrho)$.

Consider now any concept atom $A(y) \in \alpha$.

- If $\mathsf{f}(y) = \epsilon$, then we know that either (i) $A \in \mathsf{N_{RC}}$ and $A(y_\epsilon) \in \psi$, or (ii) $\mathcal{O} \models \bigsqcap \mathcal{B}|_\mathsf{R} \models A$. In both cases, it follows that $\pi'(y) = \pi(y_\epsilon) \in A^\mathcal{I}$.
- If $\mathsf{f}(y) = \varrho \cdot (R, \mathcal{C})$, then either (i) $\mathsf{f}(y)$ is rigidly witnessed in ψ', or (ii) there is a set $\mathcal{B}_A \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $\mathcal{O} \models \bigsqcap \mathcal{B}_A \sqsubseteq A$ and $\mathcal{B}_A(y_{\mathsf{f}(y)}) \subseteq \psi$. In the second case, we get $\pi'(y) = \pi(y_{\mathsf{f}(y)}) \in A^\mathcal{I}$ as above. In the first case, the above construction of π' implies that $\pi'(y) \in (\exists R^-)^\mathcal{I}$. By the third condition of $\mathsf{Con}(\alpha, \mathsf{f})$, we conclude that $\pi'(y) \in A^\mathcal{I}$.

For a role atom $S(y, y') \in \alpha$, by Definition 3.7 we have to distinguish two cases.

- If $\mathsf{f}(y) = \varrho$, $\mathsf{f}(y') = \varrho \cdot (R, \mathcal{C})$, and $\mathcal{O} \models S' \sqsubseteq S$ for some $S' \in \mathcal{C}$, then by Definition 4.7 either (i) $\mathsf{f}(y')$ is rigidly witnessed in ψ, or (ii) $S'(y_{\mathsf{f}(y)}, y_{\mathsf{f}(y')}) \in \psi$. In the first case, we know from the definition of π' that $(\pi'(y), \pi'(y')) \in R^\mathcal{I}$, and hence also $(\pi'(y), \pi'(y')) \in (S')^\mathcal{I} \subseteq S^\mathcal{I}$ by the second condition of Definition 3.7. In case (ii), we directly obtain $(\pi'(y), \pi'(y')) = (\pi(y_{\mathsf{f}(y)}), \pi(y_{\mathsf{f}(y')})) \in (S')^\mathcal{I} \subseteq S^\mathcal{I}$.
- The other case follows dual arguments, exchanging $\mathsf{f}(y)$ and $\mathsf{f}(y')$, and replacing S by S^-.

This shows that π' is a homomorphism of α into $\mathcal{I}$.

Consider now the second case of Definition 4.7, i.e., there are $R \in \mathsf{N}_\mathsf{R}^-(\mathcal{O})$, $\mathcal{B} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$, and a tree witness f for α such that $\mathcal{B}$ is a witness of $\exists R$ w.r.t. $\mathcal{O}$, $\{\exists R\} \in \mathsf{Con}(\alpha, \mathsf{f})$, and $\psi = \exists x.\mathcal{B}(x)$. From the first and the last property and the fact that $\mathcal{I} \models \psi$, we know that $\mathcal{I}$ contains an element e that satisfies $\exists R$. We define π' in analogy to the first case above, and begin by setting $\pi'(\epsilon) := e$. We can define the remainder of the homomorphism by induction on $\mathsf{range}(\mathsf{f})$ as in the case for the indirectly rigidly witnessed elements above, by treating ϵ as if it was rigidly witnessed. Only in the first step, instead of $\pi'(\epsilon)$ being an instance of some $\exists R_1^-$ in $\mathcal{I}$, we know that it satisfies $\exists R$, and hence we obtain the required role successors by the second condition of $\mathsf{Con}(\alpha, \mathsf{f})$.

Since all elements can be treated as rigidly witnessed, the remaining proof is a special case of the arguments above, except in the case of a concept atom $A(y) \in \alpha$ with $\mathsf{f}(y) = \epsilon$. But then we know that $\pi'(y) \in (\exists R)^{\mathcal{I}}$, and hence $\pi'(y) \in A^{\mathcal{I}}$ by the first condition of $\mathsf{Con}(\alpha, \mathsf{f})$. □

We can now use this lemma to prove the first direction of Lemma 4.9, namely that, if $\mathcal{S}$ is r-satisfiable w.r.t. ι and $\mathcal{O}$, then there is an r-complete tuple w.r.t. $\mathcal{S}$ and ι.

For this purpose, let $\mathcal{J}_1, \dots, \mathcal{J}_k, \mathcal{I}_0, \dots, \mathcal{I}_n$ be the interpretations over the domain Δ that exist according to the r-satisfiability of $\mathcal{S}$ (cf. Definition 4.2). We assume w.l.o.g. that Δ contains $\mathsf{N_I}$ and that all individual names are interpreted as themselves in all of these interpretations. We first define the tuple $(\mathcal{A}_\mathsf{R}, Q_\mathsf{R}, Q_\mathsf{R}^\neg)$ as follows:

$$
\begin{aligned}
\mathcal{A}_\mathsf{R} := {} & \{B(a) \mid a \in \mathsf{N_I}(\mathcal{K}), B \in \mathsf{BC_R}(\mathcal{O}), a^{\mathcal{J}_1} \in B^{\mathcal{J}_1}\} \cup \\
& \{\neg B(a) \mid a \in \mathsf{N_I}(\mathcal{K}), B \in \mathsf{BC_R}(\mathcal{O}), a^{\mathcal{J}_1} \notin B^{\mathcal{J}_1}\} \cup \\
& \{R(a,b) \mid a, b \in \mathsf{N_I}(\mathcal{K}), R \in \mathsf{N_{RR}}(\mathcal{O}), (a,b) \in R^{\mathcal{J}_1}\} \cup \\
& \{\neg R(a,b) \mid a, b \in \mathsf{N_I}(\mathcal{K}), R \in \mathsf{N_{RR}}(\mathcal{O}), (a,b) \notin R^{\mathcal{J}_1}\}; \\
Q_\mathsf{R} := {} & \{\alpha_j \in \mathcal{Q}_\phi \mid X \in \mathcal{S}, p_j \in X\}; \text{ and} \\
Q_\mathsf{R}^\neg := {} & \{\alpha_j \in \mathcal{Q}_\phi \mid X \in \mathcal{S}, p_j \notin X\}.
\end{aligned}
$$

Obviously, $\mathcal{A}_\mathsf{R}$ is an ABox type for $\mathcal{O}$, Q_R satisfies Condition (R3), and $Q_\mathsf{R}^\neg$ satisfies Condition (R4).

Our goal is to modify the given interpretations into models of the knowledge bases $\mathcal{K}_\mathsf{R}^i$, but need to ensure that the individuals of the form a_x are always interpreted by the same elements of the common domain. For this purpose, we consider the canonical models $\mathcal{I}_{\mathcal{K}_\alpha}$, $\alpha \in Q_\mathsf{R}$, for the KBs $\mathcal{K}_\alpha := \langle \mathcal{O}, \mathcal{A}_\alpha \rangle$, where $\mathcal{A}_\alpha$ is obtained by instantiating all variables x in α by the corresponding individual names $a_x \in \mathsf{N_I^{aux}}$. The goal is to add, to each $\mathcal{I}_i$ ($\mathcal{J}_i$) with $p_j \in X_{\iota(i)}$ ($p_j \in X_i$), the whole interpretation $\mathcal{I}_{\alpha_j}$, which includes a homomorphic image of α_j that involves the same domain elements (from $\mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$) in each interpretation. Note that $\mathcal{K}_\alpha$ is consistent since $\alpha \in Q_\mathsf{R}$, and hence there must be one $\mathcal{I}_i$ ($\mathcal{J}_i$) that satisfies α and thus is a model of $\mathcal{K}_\alpha$.[6]

For those interpretations $\mathcal{I}_i$ ($\mathcal{J}_i$) that do not satisfy α, we nevertheless know that they satisfy its rigid consequences $\mathcal{C}_\mathcal{O}(\alpha)$ since all these interpretations respect the rigid names and α is satisfied by at least one of them. Hence, we can also add the rigid consequences of $\mathcal{I}_{\mathcal{K}_\alpha}$ to every interpretation without losing the property that they satisfy $\mathcal{O}$ (and potentially $\mathcal{A}_i$). For this purpose, we consider similar

[6] If two variables are mapped by the homomorphism to the same domain element, we obtain a model respecting the UNA by creating a copy of this element that satisfies exactly the same concept names and participates in the same role connections as the original element.

interpretations $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_\alpha}$, which are inductively defined as in Definition 3.1, but starting instead with the following interpretation for all symbols $X \in \mathsf{N_C} \cup \mathsf{N_R}$:

$$X^0 := \begin{cases} X^{\mathcal{I}_{\mathcal{K}_\alpha}} & \text{if } X \in \mathsf{N_{RC}} \cup \mathsf{N_{RR}}, \\ \emptyset & \text{otherwise.} \end{cases}$$

This means that they must behave exactly as $\mathcal{I}_{\mathcal{K}_\alpha}$ on the rigid names, but the interpretation of the flexible names contains only those tuples that are implied by the rigid information. Each $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_\alpha}$ is a model of $\mathcal{O}$ and $\mathcal{C}_{\mathcal{O}}(\alpha)$ (see Proposition 3.2). Note that the domains of $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_\alpha}$ and $\mathcal{I}_{\mathcal{K}_\alpha}$ coincide since all elements c_ϱ that would be created by the iteration in Definition 3.1 for $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_\alpha}$ have already been created by the one for $\mathcal{I}_{\mathcal{K}_\alpha}$, and hence are already contained in the initial interpretation above.

The crucial properties of the above introduced interpretations are the following:

- $\mathcal{I}_{\mathcal{K}_{\alpha_j}}$ can be homomorphically embedded into each $\mathcal{I}_i$ ($\mathcal{J}_i$) for which we have $p_j \in X_{\iota(i)}$ ($p_j \in X_i$) since there must be a homomorphism of α_j into $\mathcal{I}_i$ ($\mathcal{J}_i$) and this entails the existence of domain elements that must satisfy at least the symbols satisfied by the elements of $\mathcal{I}_{\mathcal{K}_{\alpha_j}}$.
- $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_{\alpha_j}}$ can be homomorphically embedded into each $\mathcal{I}_i$ ($\mathcal{J}_i$), because there must be some $\mathcal{J}_x$, $1 \leq x \leq k$, that satisfies α_j and hence its rigid consequences (i.e., those of $\mathcal{I}_{\mathcal{K}_{\alpha_j}}$) are satisfied in all of these interpretations.

These facts imply that it is safe to add these new interpretations to $\mathcal{I}_i$ ($\mathcal{J}_i$), as they already contain elements that behave in exactly the same way.

We now modify the interpretations $\mathcal{I}_i$ ($\mathcal{J}_i$) into models $\mathcal{I}'_i$ ($\mathcal{J}'_i$) of $\mathcal{K}^i_{\mathsf{R}}$ ($\mathcal{K}^{n+i}_{\mathsf{R}}$), with the exception of $\mathcal{A}_{R_\mathsf{F}}$, as follows:

- The common domain Δ is extended by the union of the domains of $\mathcal{I}_{\mathcal{K}_\alpha}$, $\alpha \in Q_{\mathsf{R}}$ (which are also the domains of $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_\alpha}$). Note that these domains may overlap in $\mathsf{N_I}$.
- The individual names from $\mathsf{N_I^{aux}}$ are interpreted as themselves.
- For each j, $1 \leq j \leq m$, and $\mathcal{I}_i$ ($\mathcal{J}_i$) with $p_j \in X_{\iota(i)}$ ($p_j \in X_i$), we interpret all symbols on the domain of $\mathcal{I}_{\mathcal{K}_{\alpha_j}}$ exactly as in $\mathcal{I}_{\mathcal{K}_{\alpha_j}}$. Note that there are no role connections between the old and the new domains except between $\mathsf{N_I}(\mathcal{K})$ and $\mathsf{N_I^{aux}}$.
- If $p_j \in \overline{X_{\iota(i)}}$ ($p_j \in \overline{X_i}$), we interpret all symbols as in $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_{\alpha_j}}$.

We then have the following:

- All interpretations $\mathcal{I}'_i$ ($\mathcal{J}'_i$) satisfy $\mathcal{O}$ and $\mathcal{A}_{\mathsf{R}}$ since the new domain elements do not exhibit new behavior that was not already present in $\mathcal{I}_i$ ($\mathcal{J}_i$) and the interpretation of basic concepts on the elements of $\mathsf{N_I}$ does not change.
- Each $\mathcal{I}'_i$ still satisfies $\mathcal{A}_i$ because of the same reason.
- Each interpretation is a model of $\mathcal{A}_{Q_{\mathsf{R}}}$ since that ABox consists exactly of the ABoxes $\mathcal{C}_{\mathcal{O}}(\alpha)$, $\alpha \in Q_{\mathsf{R}}$, which are satisfied by the new domain elements because of $\mathcal{I}^{\mathsf{R}}_{\mathcal{K}_\alpha}$, which is contained in $\mathcal{I}_{\mathcal{K}_\alpha}$.
- Each $\mathcal{I}'_i$ ($\mathcal{J}'_i$) satisfies $\mathcal{A}_{Q_{\iota(i)}}$ ($\mathcal{A}_{Q_i}$) since they consist exactly of the ABoxes $\mathcal{A}_{\alpha_j}$ with $p_j \in X_{\iota(i)}$ ($p_j \in X_i$), which are satisfied by $\mathcal{I}_{\mathcal{K}_{\alpha_j}}$.
- For each $\mathcal{I}_i$ ($\mathcal{J}_i$) and $p_j \in \overline{X_{\iota(i)}}$ ($p_j \in \overline{X_i}$), we know that $\mathcal{I}'_i \not\models \alpha_j$ ($\mathcal{J}'_i \not\models \alpha_j$) since any homomorphism of α_j into $\mathcal{I}'_i$ ($\mathcal{J}'_i$) would allow us to find one into $\mathcal{I}_i$ ($\mathcal{J}_i$) as well, which contradicts the assumption that $\mathcal{I}_i \models \chi_{\iota(i)}$ ($\mathcal{J}_i \models \chi_i$). Hence, we know that $\mathcal{I}'_i \models \chi_{\iota(i)}$ ($\mathcal{J}'_i \models \chi_i$).
- All interpretations $\mathcal{I}'_0, \dots, \mathcal{I}'_n$ and $\mathcal{J}'_1, \dots, \mathcal{J}'_k$ respect the rigid names.

If ϕ is not r-simple, then we also have to define R_{F} and extend the above interpretations to models of $\mathcal{A}_{R_{\mathsf{F}}}$:

$$\begin{aligned} R_{\mathsf{F}} := \{\exists S(b) \mid\ & S \in \mathsf{N_R}(\mathcal{O}) \setminus \mathsf{N_{RR}},\ b \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}, \\ & i \in \{0, \dots, n+k\},\ \langle \mathcal{O}, \mathcal{A}_{\mathsf{R}} \cup \mathcal{A}_{Q_{\mathsf{R}}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_i \rangle \models \exists S(b)\}. \end{aligned}$$

Since the interpretations $\mathcal{I}'_i$ ($\mathcal{J}'_i$) are models of these knowledge bases, for each $\exists S(b) \in R_{\mathsf{F}}$ we know that one of these models satisfies the assertion. Hence, the rigid consequences described in $\mathcal{A}_{R_{\mathsf{F}}}$ must already be satisfied by some domain elements in the common domain (in all of these interpretations). We can define the interpretation of the elements $a_{b\varrho} \in \mathsf{N_I^{tree}}$ accordingly, but again may have to copy elements if the UNA would be violated otherwise. Note that Condition (R6) is obviously satisfied by our definition of R_{F}. These modified interpretations $\mathcal{I}'_i$ ($\mathcal{J}'_i$) prove now that Condition (R1) is also satisfied.

Regarding Condition (R2), assume that there are an index i, $0 \leq i \leq n$ (or $n+1 \leq i \leq n+k$), and $p_j \in \overline{X_{\iota(i)}}$ such that $\mathcal{K}^i_{\mathsf{R}} \models \alpha_j$, and thus $\mathcal{I}'_i \models \alpha_j$ ($\mathcal{J}'_{i-n} \models \alpha_j$). But this contradicts the fact that $\mathcal{I}'_i \models \chi_{\iota(i)}$ ($\mathcal{J}'_{i-n} \models \chi_{i-n}$).

The proof of Condition (R5) is also by contradiction. We assume that there are i, $0 \leq i \leq n$ ($n+1 \leq i \leq n+k$), a tree-shaped CQ $\alpha_j \in Q^{\neg}_{\mathsf{R}}$, and a witness query ψ for α_j w.r.t. $\mathcal{K}^i_{\mathsf{R}}$ such that $\mathcal{K}^i_{\mathsf{R}} \models \psi$, and thus $\mathcal{I}'_i \models \psi$ ($\mathcal{J}'_{i-n} \models \psi$). However, $\alpha_j \in Q^{\neg}_{\mathsf{R}}$ yields that there is an $X_x \in \mathcal{S}$, $1 \leq x \leq k$, such that $p_j \notin X_x$, and thus $\mathcal{J}'_x \not\models \alpha_j$. By Lemma 4.10, we know that $\mathcal{J}'_x \not\models \psi$. But this contradicts the facts that ψ contains only rigid names and $\mathcal{J}'_x$ and $\mathcal{I}'_i$ ($\mathcal{J}'_{i-n}$) respect the rigid names. □

This concludes the proof of the first direction of Lemma 4.9.

4.2.4 If there is an r-complete tuple w.r.t. $\mathcal{S}$ and ι, then $\mathcal{S}$ is r-satisfiable w.r.t. ι and $\mathcal{O}$.

The proof of the converse direction is more involved. We assume an r-complete tuple $(\mathcal{A}_{\mathsf{R}}, Q_{\mathsf{R}}, Q_{\mathsf{R}}^{\neg}, R_{\mathsf{F}})$ to be given. Further, if ϕ is r-simple, then we can assume R_{F}, and hence $\mathcal{A}_{R_{\mathsf{F}}}$ and $\mathsf{N}_{\mathsf{I}}^{\mathsf{tree}}$, to be empty.

For each i, $0 \leq i \leq n+k$, we consider the canonical interpretation $\mathcal{I}_i := \mathcal{I}_{\mathcal{K}_{\mathsf{R}}^i}$. To distinguish the elements contained in $\mathsf{N}_{\mathsf{I}}^{\mathsf{aux}}$, we write a_x^i for the element $a_x \in \mathsf{N}_{\mathsf{I}}^{\mathsf{aux}}$ in the domain of $\mathcal{I}_i$, and define the set $\Delta_{\mathsf{a}}^{\mathcal{I}_i}$ to contain exactly those elements. Likewise, we write $a_{b\varrho}^i$ for the element $a_{b\varrho} \in \mathsf{N}_{\mathsf{I}}^{\mathsf{tree}}$ that occurs in $\mathcal{I}_i$, and define $\Delta_{\mathsf{t}}^{\mathcal{I}_i}$ as above. We further write $\Delta_{\mathsf{u}}^{\mathcal{I}_i}$ for the set containing the unnamed domain elements unique to the canonical interpretation $\mathcal{I}_i$, and similarly write $c_{\varrho R}^i$ for every element $c_{\varrho R} \in \Delta_{\mathsf{u}}^{\mathcal{I}_i}$. For any $e \in \mathsf{N}_{\mathsf{I}}(\mathcal{K}) \cup \mathsf{N}_{\mathsf{I}}^{\mathsf{aux}} \cup \mathsf{N}_{\mathsf{I}}^{\mathsf{tree}}$, we may denote by e^i the corresponding element in $\mathsf{N}_{\mathsf{I}}(\mathcal{K}) \cup \Delta_{\mathsf{a}}^{\mathcal{I}_i} \cup \Delta_{\mathsf{t}}^{\mathcal{I}_i}$, i.e., we consider $a^i := a$ for any $a \in \mathsf{N}_{\mathsf{I}}(\mathcal{K})$. Thus, the domain of each $\mathcal{I}_i$ is composed of the pairwise disjoint components $\mathsf{N}_{\mathsf{I}}(\mathcal{K})$, $\Delta_{\mathsf{a}}^{\mathcal{I}_i}$, $\Delta_{\mathsf{t}}^{\mathcal{I}_i}$, and $\Delta_{\mathsf{u}}^{\mathcal{I}_i}$. We state this fact for future reference.

Fact 4.11. *For all $i, j, j' \in \{0, \ldots, n+k\}$, the sets $\mathsf{N}_{\mathsf{I}}(\mathcal{K})$, $\Delta_{\mathsf{a}}^{\mathcal{I}_i}$, $\Delta_{\mathsf{t}}^{\mathcal{I}_j}$, and $\Delta_{\mathsf{u}}^{\mathcal{I}_{j'}}$ are pairwise disjoint.*

We now construct the interpretations $\mathcal{J}_0, \ldots, \mathcal{J}_{n+k}$ required for the r-satisfiability of $\mathcal{S}$ (where $\mathcal{J}_0, \ldots, \mathcal{J}_n$ represent $\mathcal{I}_0, \ldots, \mathcal{I}_n$ of Definition 4.2 and $\mathcal{J}_{n+1}, \ldots, \mathcal{J}_{n+k}$ represent $\mathcal{J}_1, \ldots, \mathcal{J}_k$). To this end, we join the domains of the interpretations $\mathcal{I}_i$ and ensure that they interpret all rigid names in the same way. We first construct the common domain

$$\Delta := \mathsf{N}_{\mathsf{I}}(\mathcal{K}) \cup \bigcup_{i=0}^{n+k} (\Delta_{\mathsf{a}}^{\mathcal{I}_i} \cup \Delta_{\mathsf{t}}^{\mathcal{I}_i} \cup \Delta_{\mathsf{u}}^{\mathcal{I}_i})$$

and then define the interpretations $\mathcal{J}_i$, $0 \leq i \leq n+k$, as follows:

- For all $a \in \mathsf{N}_{\mathsf{I}}(\mathcal{K})$, we set $a^{\mathcal{J}_i} := a$.
- For all rigid concept names A, we define $A^{\mathcal{J}_i} := \bigcup_{j=0}^{n+k} A^{\mathcal{I}_j}$.
- For all flexible concept names A, we define

$$A^{\mathcal{J}_i} := A^{\mathcal{I}_i} \cup \bigcup_{j=0}^{n+k} \bigcup_{\substack{\mathcal{B} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O}), \\ \mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq A}} (\bigsqcap \mathcal{B})^{\mathcal{I}_j} \cup$$
$$\bigcup_{j=0}^{n+k} \{c_{\varrho R}^j \in \Delta_{\mathsf{u}}^{\mathcal{I}_j} \mid \mathcal{O} \models \exists R^- \sqsubseteq A, \mathcal{W}_{\mathcal{O}}(c_{\varrho R}^j) \neq \emptyset\}.$$

- For all rigid role names R, we define $R^{\mathcal{J}_i} := \bigcup_{j=0}^{n+k} R^{\mathcal{I}_j}$.

- For all flexible role names R, we define

$$R^{\mathcal{J}_i} := R^{\mathcal{I}_i} \cup \bigcup_{j=0}^{n+k} \bigcup_{\substack{S \in \mathsf{N}_{\mathsf{RR}}^-(\mathcal{O}) \\ \mathcal{O} \models S \sqsubseteq R}} S^{\mathcal{I}_j} \cup \bigcup_{j=0}^{n+k} \{(e_1, e_2) \in R^{\mathcal{I}_j} \mid \exists e \in \{e_1, e_2\}, \mathcal{W}_{\mathcal{O}}(e) \neq \emptyset\}.$$

In this way, we have constructed interpretations $\mathcal{J}_0, \dots, \mathcal{J}_{n+k}$ that have the same domain and respect rigid names.

It remains to show that the interpretations satisfy the other requirements for the r-satisfiability of $\mathcal{S}$. To this end, we first provide auxiliary lemmas and begin showing a basic connection between the interpretations $\mathcal{J}_i$ and $\mathcal{I}_i$ concerning the interpretation of role names.

Lemma 4.12. *For all $i \in \{0, \dots, n+k\}$, role names $R \in \mathsf{N}_\mathsf{R}$, and $d, e \in \Delta^{\mathcal{I}_i}$, we have $(d, e) \in R^{\mathcal{J}_i}$ iff $(d, e) \in R^{\mathcal{I}_i}$.*

Proof. The "if"-direction follows directly from the definition of $R^{\mathcal{J}_i}$. We consider the "only if"-direction.

If R is flexible, assume that either (i) $(d, e) \in S^{\mathcal{I}_j}$ for some rigid subrole S of R or (ii) $(d, e) \in R^{\mathcal{I}_j}$, $j \neq i$, and either d or e has a witness w.r.t. $\mathcal{O}$. By Fact 4.11, in both cases we must have $d, e \in \mathsf{N}_\mathsf{I}(\mathcal{K})$. Hence, case (ii) is impossible since named domain elements cannot have witnesses (see Definition 4.5). In case (i), we must have $S(d, e) \in \mathcal{A}_\mathsf{R}$ since $\mathcal{A}_\mathsf{R}$ is an ABox type and $\mathcal{I}_j \models \mathcal{A}_\mathsf{R}$, and hence $(d, e) \in S^{\mathcal{I}_i} \subseteq R^{\mathcal{I}_i}$ since $\mathcal{I}_i \models \mathcal{A}_\mathsf{R}$ and $\mathcal{I}_i \models \mathcal{O}$.

If R is rigid, assume again that $d, e \in \mathsf{N}_\mathsf{I}(\mathcal{K})$ and $(d, e) \in R^{\mathcal{I}_j}$, for some $j \neq i$. Since $\mathcal{A}_\mathsf{R}$ is an ABox type and $\mathcal{I}_j \models \mathcal{A}_\mathsf{R}$, we must have $R(d, e) \in \mathcal{A}_\mathsf{R}$. Since also $\mathcal{I}_i \models \mathcal{A}_\mathsf{R}$, we get $(d, e) \in R^{\mathcal{I}_i}$. □

There is a similar connection between the interpretations of concepts in $\mathcal{J}_i$ and $\mathcal{I}_j$.

Lemma 4.13. *For all basic concepts $B \in \mathsf{BC}(\mathcal{O})$ and $i, j \in \{0, \dots, n+k\}$, the following hold:*

a) *for all $a \in \mathsf{N}_\mathsf{I}(\mathcal{K})$, we have $a \in B^{\mathcal{J}_i}$ iff $a \in B^{\mathcal{I}_i}$;*

b) *if B is rigid, then, for every $e \in \Delta_\mathsf{a}^{\mathcal{I}_j} \cup \Delta_\mathsf{t}^{\mathcal{I}_j} \cup \Delta_\mathsf{u}^{\mathcal{I}_j}$, we have $e \in B^{\mathcal{J}_i}$ iff $e \in B^{\mathcal{I}_j}$;*

c) *if B is flexible, then, for every $e \in \Delta_\mathsf{a}^{\mathcal{I}_j} \cup \Delta_\mathsf{t}^{\mathcal{I}_j} \cup \Delta_\mathsf{u}^{\mathcal{I}_j}$, we have $e \in B^{\mathcal{J}_i}$ iff*

- *$i = j$ and $e \in B^{\mathcal{I}_i}$, or*
- *there is a $\mathcal{B} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $e \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ and $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq B$, or*
- *$e \in B^{\mathcal{I}_j} \cap \Delta_\mathsf{u}^{\mathcal{I}_j}$ and $\mathcal{W}_\mathcal{O}(e) \neq \emptyset$.*

Proof. For a), consider first the case that B is rigid. Then, $a \in B^{\mathcal{I}_i}$ clearly implies $a \in B^{\mathcal{J}_i}$ since $s^{\mathcal{I}_i} \subseteq s^{\mathcal{J}_i}$, for $s \in \mathsf{N}_\mathsf{RC} \cup \mathsf{N}_\mathsf{RR}$. On the other hand, if $a \in B^{\mathcal{J}_i}$, then by the definition of $\mathcal{J}_i$, we must have $a \in B^{\mathcal{I}_j}$, for at least one $\mathcal{I}_j$, $0 \leq j \leq n+k$. Since $\mathcal{I}_i$ and $\mathcal{I}_j$ are both models of the ABox type $\mathcal{A}_\mathsf{R}$, we also have $a \in B^{\mathcal{I}_i}$.

Consider now any flexible basic concept B. By the definition of $\mathcal{J}_i$ on the flexible names, we have $a \in B^{\mathcal{J}_i}$ iff either (i) $a \in B^{\mathcal{I}_i}$, or (ii) $a \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$, for some j, $0 \leq j \leq n+k$, and $\mathcal{B} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq B$. To see the latter w.r.t. a flexible concept of the form $B = \exists R$, note that the definition of $R^{\mathcal{J}_i}$ implies one of the following cases (assuming that case (i) does not apply):

- a belongs to $(\exists S)^{\mathcal{I}_j}$, where S is a rigid subrole of R. Hence, we can choose $\mathcal{B} := \{\exists S\}$ since also $\exists S \sqsubseteq \exists R$.
- There is an R-successor of a in $\mathcal{J}_i$ of the form $c^j_{aR_2}$ for some $R_2 \in \mathsf{N}^-_\mathsf{R}(\mathcal{O})$ with $\mathcal{O} \models R_2 \sqsubseteq R$. By Definition 4.5, $\mathcal{W}_\mathcal{O}(a)$ is not defined, and hence there must exist a witness $\mathcal{B} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ such that $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R_2 \sqsubseteq \exists R$.

But (ii) implies (i) since $a \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ yields $B'(a) \in \mathcal{A}_\mathsf{R}$, for all $B' \in \mathcal{B}$, which together with $\mathcal{I}_i \models \mathcal{A}_\mathsf{R}$ and $\mathcal{I}_i \models \mathcal{O}$ leads to $a \in B^{\mathcal{I}_i}$.

The claim in b) follows directly from Fact 4.11 and the definition of $\mathcal{J}_i$.

For c), we first consider the case that $B \in \mathsf{N}_\mathsf{C}(\mathcal{O})$ is a flexible concept name. Then, the equivalence with one of the three cases is covered by the definition of $\mathcal{J}_i$ and, for the last case, also by Proposition 3.4. We now consider B to be of the form $\exists R$ for a flexible role $R \in \mathsf{N}^-_\mathsf{R}(\mathcal{O})$. In case $i = j$, the claim can be restricted to the first of the three items since the other two are subsumed by it (for the second item, this holds because $\mathcal{I}_i \models \mathcal{O}$). Hence, the claim directly follows from Fact 4.11 and the definition of $\mathcal{J}_i$ (i.e., the interpretation of the elements in $\Delta_\mathsf{a}^{\mathcal{I}_i} \cup \Delta_\mathsf{t}^{\mathcal{I}_i} \cup \Delta_\mathsf{u}^{\mathcal{I}_i}$ is not influenced by any $\mathcal{I}_j$, $j \neq i$). We consider the case $i \neq j$.

- Let e be of the form d^j for some $d \in \mathsf{N}_\mathsf{I}^\mathsf{aux} \cup \mathsf{N}_\mathsf{I}^\mathsf{tree}$, i.e., $e \in \Delta_\mathsf{a}^{\mathcal{I}_j} \cup \Delta_\mathsf{t}^{\mathcal{I}_j}$. We only have to regard the second item. ($\Rightarrow$) The definition of $R^{\mathcal{J}_i}$ yields that either (i) $e \in (\exists S)^{\mathcal{J}_i}$ and $\mathcal{O} \models S \sqsubseteq R$ for a rigid role S or (ii) there is an R-successor e' of e in $\mathcal{I}_j$ and either e or e' has a witness w.r.t. $\mathcal{O}$. In case (i), we can choose $\mathcal{B} := \{\exists S\}$. In case (ii), since $\mathcal{W}_\mathcal{O}$ is not defined for elements of $\mathsf{N}_\mathsf{I}(\mathcal{K}) \cup \Delta_\mathsf{a}^{\mathcal{I}_j} \cup \Delta_\mathsf{t}^{\mathcal{I}_j}$, we know that e' is of the form $c^j_{dR_2}$ for some $R_2 \in \mathsf{N}^-_\mathsf{R}(\mathcal{O})$ and $\mathcal{W}_\mathcal{O}(c^j_{dR_2}) \neq \emptyset$. By Definition 3.1, we obtain $\mathcal{O} \models R_2 \sqsubseteq R$. By the definition of $\mathcal{W}_\mathcal{O}$, there is a $\mathcal{B} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $e \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ and

$\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R_2 \sqsubseteq \exists R$. ($\Leftarrow$) We have $e \in (\exists R)^{\mathcal{I}_j}$, and hence $c^j_{dR} \in \Delta^{\mathcal{I}_j}_{\mathsf{u}}$ and $(e, c^j_{dR}) \in R^{\mathcal{I}_j}$, by Definition 3.1. Furthermore, $\mathcal{B}$ is a witness of c^j_{dR}, and thus we have $(e, c^j_{dR}) \in R^{\mathcal{J}_i}$ by the definition of $\mathcal{J}_i$.

- Let $e \in \Delta^{\mathcal{I}_j}_{\mathsf{u}}$. ($\Rightarrow$) We know that (i) $e \in (\exists S)^{\mathcal{I}_j}$ and $\mathcal{O} \models S \sqsubseteq R$ for a rigid role S (hence we can again choose $\mathcal{B} := \{\exists S\}$), or (ii) $(e, d) \in R^{\mathcal{I}_j}$ for some $d \in \Delta^{\mathcal{I}_j}$ (and hence also $e \in (\exists R)^{\mathcal{I}_j}$), and either $\mathcal{W}_{\mathcal{O}}(e)$ or $\mathcal{W}_{\mathcal{O}}(d)$ is non-empty. Thus, if $\mathcal{W}_{\mathcal{O}}(d)$ is undefined or empty, then the third item holds. Otherwise, we know that $d \in \Delta^{\mathcal{I}_j}_{\mathsf{u}}$ and $\mathcal{W}_{\mathcal{O}}(d) \neq \emptyset$, by Definition 4.5. By Definition 3.1, we then have either (i) $e = c^j_{\varrho}$ and $d = c^j_{\varrho R}$; or (ii) $d = c^j_{\varrho}$ and $e = c^j_{\varrho R^-}$.

  For (i), Definition 4.5 yields that either $\mathcal{W}_{\mathcal{O}}(e)$ is also non-empty (the third item), or there is a $\mathcal{B} \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ such that $e \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ and $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R$ (the second item).

  For (ii), we immediately have that the witness of d is also a witness of e, again by Definition 4.5.

  ($\Leftarrow$) Let $e = c^j_{\varrho}$. If there is a set $\mathcal{B} \in \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ with $e \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ and $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R$, then by Definition 3.1, the element $c^j_{\varrho R} \in \Delta^{\mathcal{I}_j}_{\mathsf{u}}$ exists and $(e, c^j_{\varrho R}) \in R^{\mathcal{I}_j}$. Furthermore, $\mathcal{B}$ is a witness of $c^j_{\varrho R}$, and hence $(e, c^j_{\varrho R}) \in R^{\mathcal{J}_i}$, i.e., $e \in (\exists R)^{\mathcal{J}_i}$, by the definition of $\mathcal{J}_i$.

  If $e \in (\exists R)^{\mathcal{I}_j}$, then we also have $(e, c^j_{\varrho R}) \in R^{\mathcal{I}_j}$, by Definition 3.1. Moreover, $\mathcal{W}_{\mathcal{O}}(e) \neq \emptyset$ yields $e \in (\exists R)^{\mathcal{J}_i}$, as in the previous case. ☐

We finally show that $\mathcal{J}_i$ is in fact as intended.

Lemma 4.14. *Each $\mathcal{J}_i$, $0 \leq i \leq n + k$, is a model of $(\mathcal{O}, \mathcal{A}_i)$.*

Proof. For any assertion $\alpha \in \mathcal{A}_i$, we have $\mathcal{I}_i \models \alpha$, and thus Lemmas 4.12 and 4.13a) yield $\mathcal{J}_i \models \alpha$.

Consider a CI $B_1 \sqcap \cdots \sqcap B_m \sqsubseteq B \in \mathcal{O}$, where $B_1, \ldots, B_m$ are basic concepts and B is either a basic concept or $\bot$. Let $d \in B_1^{\mathcal{J}_i} \cap \cdots \cap B_m^{\mathcal{J}_i}$. If $d \in \mathsf{N_I}(\mathcal{K})$, we get $d \in B_1^{\mathcal{I}_i} \cap \cdots \cap B_m^{\mathcal{I}_i}$ by Lemma 4.13a). Since $\mathcal{I}_i \models \mathcal{O}$, this implies that $d \in B^{\mathcal{I}_i}$. If $B = \bot$, this is impossible. Otherwise, we get $d \in B^{\mathcal{J}_i}$, again by Lemma 4.13a).

Consider now the case that $d \in \Delta^{\mathcal{I}_j}_{\mathsf{a}} \cup \Delta^{\mathcal{I}_j}_{\mathsf{t}} \cup \Delta^{\mathcal{I}_j}_{\mathsf{u}}$. If $i = j$, then Lemma 4.13b) and Lemma 4.13c) yield the same conclusion as above since the latter collapses to the first item. Assume now that $i \neq j$. By Lemma 4.13, for every B_ℓ, $1 \leq \ell \leq m$, we have that (i) there is a $\mathcal{B}_\ell \subseteq \mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ with $d \in (\bigsqcap \mathcal{B}_\ell)^{\mathcal{I}_j}$ and $\mathcal{O} \models \bigsqcap \mathcal{B}_\ell \sqsubseteq B_\ell$ (in case B_ℓ is rigid, we can set $\mathcal{B}_\ell := \{B_\ell\}$); or (ii) $d \in B_\ell^{\mathcal{I}_j} \cap \Delta^{\mathcal{I}_j}_{\mathsf{u}}$ and $\mathcal{W}_{\mathcal{O}}(d) \neq \emptyset$. Since $\mathcal{I}_j \models \mathcal{O}$, in either case, we know that $d \in B_\ell^{\mathcal{I}_j}$, and hence $d \in B^{\mathcal{I}_j}$. If $B = \bot$, this is again impossible. If B is rigid, then Lemma 4.13b) yields $d \in B^{\mathcal{J}_i}$, as required. If B is flexible and case (ii) applies for at least one B_ℓ, then the third item of Lemma 4.13c) yields the claim. Otherwise, it is easy to see that for $\mathcal{B} := \bigcup_{\ell=1}^{m} \mathcal{B}_\ell$ we have $d \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ and $\mathcal{O} \models \mathcal{B} \sqsubseteq B_1 \sqcap \cdots \sqcap B_m \sqsubseteq B$. Hence, the second item of Lemma 4.13c) applies, and we also get $d \in B^{\mathcal{J}_i}$.

We now consider role inclusions of the form $R_1 \sqsubseteq R_2$ and assume that $(d, e) \in R_1^{\mathcal{J}_i}$. By the definition of $\mathcal{J}_i$, we must have $(d, e) \in R_1^{\mathcal{I}_j}$ for some $j \in \{0, \ldots, n + k\}$. Since $\mathcal{I}_j \models \mathcal{O}$, we get $(d, e) \in R_2^{\mathcal{I}_j}$. If R_2 is rigid, we immediately get $(d, e) \in R_2^{\mathcal{J}_i}$. If R_2 is flexible, assume that $(d, e) \notin R_2^{\mathcal{J}_i}$. But then we must have $i \neq j$, R_2 cannot have a rigid subrole S with $(d, e) \in S^{\mathcal{I}_j}$, and neither d nor e can have a witness w.r.t. $\mathcal{O}$. This implies that also R_1 cannot be rigid and cannot have a rigid subrole S with $(d, e) \in S^{\mathcal{I}_j}$ (since this would also be a rigid subrole of R_2). Hence, the definition of $\mathcal{J}_i$ yields that $(d, e) \notin R_1^{\mathcal{I}_j}$, which contradicts our assumption. □

We now provide the final missing piece to show r-satisfiability of $\mathcal{S}$.

Lemma 4.15. *Each $\mathcal{J}_i$, $1 \leq i \leq n + k$, is a model of χ_i.*

Proof. Consider first any CQ α that occurs positively in the conjunction χ_i. Since $\mathcal{I}_i \models \mathcal{A}_{Q_i}$ and $\mathcal{A}_{Q_i}$ contains an instantiation of α, we know that there is a homomorphism π of α into $\mathcal{I}_i$ that maps all variables to elements in $\Delta_{\mathsf{a}}^{\mathcal{I}_i}$. By Lemmas 4.12 and 4.13, we know that π is also a homomorphism of α into $\mathcal{J}_i$.

We now consider a CQ α that occurs negatively in χ_i, and assume to the contrary that there is a homomorphism π of α into $\mathcal{J}_i$. By Condition (R2), we know that $\mathcal{K}_{\mathsf{R}}^{i} \not\models \alpha$, and thus $\mathcal{I}_i \models \neg\alpha$, by Proposition 3.5. Furthermore, by (R4) we know that $\alpha \in Q_{\mathsf{R}}^{-}$.

We distinguish two cases.

(I) Let first π be such that it maps no terms into $\mathsf{N_I}(\mathcal{K}) \cup \bigcup_{j=0}^{n+k} \Delta_{\mathsf{a}}^{\mathcal{I}_j} \cup \Delta_{\mathsf{t}}^{\mathcal{I}_j}$. This in particular implies that $\mathsf{N_I}(\alpha) = \emptyset$ since α does not contain any names from $\mathsf{N_I^{aux}}$ or $\mathsf{N_I^{tree}}$. Because α is connected and by the interpretation of roles in $\mathcal{J}_i$, which is based on the canonical interpretations (cf. Definition 3.1), and Fact 4.11, we have $\mathsf{range}(\pi) \subseteq \Delta^{\mathcal{I}_j}$, for a fix j. Given that $\mathcal{I}_i \models \neg\alpha$, we directly get a contradiction if $j = i$, by Lemmas 4.12 and 4.13, and in the following assume that $j \neq i$.

By considering how the elements in $\Delta_{\mathsf{u}}^{\mathcal{I}_j}$ are connected by roles within $\mathcal{I}_j$, it is easy to see that there must be a variable $x \in \mathsf{N_V}(\alpha)$, for which $\pi(x) = c_{\varrho R}^{j}$ is such that the length of ϱR is minimal among all elements of $\mathsf{range}(\pi)$. Furthermore, all other $\pi(y)$ for $y \in \mathsf{N_V}(\alpha)$ must then be of the form $c_{\varrho R \varrho'}^{j}$ for some $\varrho' \in (\mathsf{N_R^-})^*$.

We now define a witness query based on a tree witness f for x in α. We start with a mapping $\mathsf{f}' \colon \mathsf{range}(\pi) \to (\mathsf{N_R^-} \times 2^{\mathsf{N_R^-}})^*$ and then set $\mathsf{f}(y) := \mathsf{f}'(\pi(y))$ for all $y \in \mathsf{N_V}(\alpha)$. We first define $\mathsf{f}'(c_{\varrho R}^{j}) := \epsilon$, and proceed by induction on the structure of Δ_{u}^{j}. Let $c_{\varrho R \varrho' R'}^{j}$ be such that $\mathsf{f}'(c_{\varrho R \varrho'}^{j})$ has already been defined. Then

$\mathsf{f}'(c^j_{\varrho R\varrho' R'}) := \mathsf{f}'(c^j_{\varrho R\varrho'})\cdot(R',\mathcal{C})$, where $\mathcal{C}$ is constructed as follows: for all role atoms $S(y,y')\in\alpha$ such that $\pi(y)=c^j_{\varrho R\varrho'}$ and $\pi(y')=c^j_{\varrho R\varrho' R'}$, do the following:

- if there is a $S'\in\mathsf{N}^-_{\mathsf{RR}}(\mathcal{O})$ with $\mathcal{O}\models S'\sqsubseteq S$ and $(\pi(y),\pi(y'))\in(S')^{\mathcal{J}_i}$, then add S' to $\mathcal{C}$;
- otherwise, add S to $\mathcal{C}$ (since π is a homomorphism of α into $\mathcal{J}_i$, we know that $(\pi(y),\pi(y'))\in S^{\mathcal{J}_i}$).

It follows from Definition 3.1 that in these cases we must have $\mathcal{O}\models R'\sqsubseteq S'$ or $\mathcal{O}\models R'\sqsubseteq S$, respectively, i.e., f is indeed a tree witness for x in α.

To construct a witness query ψ for α, the next step is to find a set $\mathcal{B}\in\mathsf{Con}(\alpha,\mathsf{f})$. We first verify the last two properties of Definition 3.7, which do not depend on the particular choice of $\mathcal{B}$. First, let $A(y)$ be an atom of α with $\mathsf{f}(y)=\varrho'\cdot(S,\mathcal{C})$, which implies that $\pi(y)=c^j_{\varrho R\varrho'|_1 S}\in A^{\mathcal{J}_i}$, where $\varrho'|_1$ is the projection of the sequence ϱ' on its first component. By Lemma 4.13, we know that $c^j_{\varrho R\varrho'|_1 S}\in A^{\mathcal{I}_j}$, and hence $\mathcal{O}\models\exists S^-\sqsubseteq A$ by Proposition 3.4. Second, consider $\varrho'\cdot(S_1,\mathcal{C}_1)\cdot(S_2,\mathcal{C}_2)\in\mathsf{range}(\mathsf{f})$. We know that $(c^j_{\varrho R\varrho'|_1 S_1},c^j_{\varrho R\varrho'|_1 S_1 S_2})\in S_2^{\mathcal{I}_j}$, and hence $c^j_{\varrho R\varrho'|_1 S_1}\in(\exists S_2)^{\mathcal{I}_j}$. Again by Proposition 3.4, we obtain $\mathcal{O}\models\exists S_1^-\sqsubseteq\exists S_2$.

To construct $\mathcal{B}\in\mathsf{Con}(\alpha,\mathsf{f})$, we now consider all atoms of the form $A(y)$ or $S(y,z)$ in α such that $\pi(y)=c^j_{\varrho R}$, i.e., $\mathsf{f}(y)=\epsilon$. For each concept atom $A(y)$, we set $B_{A(y)}:=A$ and observe that $c^j_{\varrho R}\in B^{\mathcal{J}_i}_{A(y)}$ since π is a homomorphism of α into $\mathcal{J}_i$. For the role atoms $S(y,z)$, we must similarly have $\pi(z)=c^j_{\varrho RS_2}$, $(\pi(y),\pi(z))\in S^{\mathcal{J}_i}$, $\mathsf{f}_x(z)=(S_2,\mathcal{C})$, and $S'\in\mathcal{C}$ for some $S_2,S'\in\mathsf{N}^-_{\mathsf{R}}(\mathcal{O})$ with $\mathcal{O}\models S_2\sqsubseteq S'\sqsubseteq S$, where we know that if $S'\neq S$, then S' is rigid. We set $B_{S(y,z)}:=\exists S_2$ and consider two cases.

- If S' is rigid and S_2 is flexible, we define $\mathcal{B}_{S(y,z)}:=\{\exists S_2\}$. Notice that $\mathcal{O}\models\bigsqcap\mathcal{B}_{S(y,z)}\sqsubseteq B_{S(y,z)}$ and $c^j_{\varrho R}\in(\bigsqcap\mathcal{B}_{S(y,z)})^{\mathcal{I}_j}$ by Proposition 3.4.
- If S' is flexible, then we know that $S'=S$ and S_2 is also flexible, and there can be no rigid role between S_2 and S. Hence, we obtain from the definition of $\mathcal{J}_i$ that either $\pi(y)$ or $\pi(z)$ must have a witness w.r.t. $\mathcal{O}$, and hence $(\pi(y),\pi(z))\in S_2^{\mathcal{J}_i}$.
- If S_2 is rigid, then we immediately get $(\pi(y),\pi(z))\in S_2^{\mathcal{I}_j}\subseteq S_2^{\mathcal{J}_i}$.

To summarize, in the cases for which $\mathcal{B}_\beta$ is not (yet) defined, we know that $\pi(y)=c^j_{\varrho R}\in B^{\mathcal{J}_i}_\beta$. From Lemma 4.13, we obtain that then either (i) there is a $\mathcal{B}_\beta\subseteq\mathsf{BC}_{\mathsf{R}}(\mathcal{O})$ with $c^j_{\varrho R}\in(\bigsqcap\mathcal{B}_\beta)^{\mathcal{I}_j}$ and $\mathcal{O}\models\bigsqcap\mathcal{B}_\beta\sqsubseteq B_\beta$, or (ii) $c^j_{\varrho R}\in B^{\mathcal{I}_j}_\beta$ and $\mathcal{W}_{\mathcal{O}}(c^j_{\varrho R})\neq\emptyset$.

- If $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) \neq \emptyset$, by Definitions 4.5 and 3.1 and Proposition 3.3, there must be a set $\mathcal{B} \subseteq \mathsf{BC_R}(\mathcal{O})$ that is a witness of $\exists R^-$ w.r.t. $\mathcal{O}$, and furthermore an element $c^j_{\varrho'} \in \Delta^j_\mathsf{u}$ that satisfies $\bigsqcap \mathcal{B}$ in $\mathcal{I}_j$. It remains to show that $\{\exists R^-\} \in \mathsf{Con}(\alpha, \mathsf{f})$. But for all $A(y) \in \alpha$ with $\mathsf{f}_x(y) = \epsilon$, we have $\mathcal{O} \models \exists R^- \sqsubseteq A$ by Proposition 3.4, because $c^j_{\varrho R} \in B^{\mathcal{I}_j}_{A(y)} = A^{\mathcal{I}_j}$ (this follows in both cases (i) and (ii) above). Likewise, for an element $(R', \mathcal{C}) \in (\mathsf{N_R^-} \times 2^{\mathsf{N_R^-}}) \cap \mathsf{range}(\mathsf{f}_x)$, we know that $c^j_{\varrho RR'} \in \Delta^j_\mathsf{u}$, and hence $c^j_{\varrho R} \in (\exists R')^{\mathcal{I}_j}$ and $\mathcal{O} \models \exists R^- \sqsubseteq \exists R'$ by Proposition 3.4. Hence, we have found a witness query $\psi := \exists x.\mathcal{B}(x)$ for α. Moreover, we have $\mathcal{I}_j \models \psi$ via the homomorphism that maps x to $c^j_{\varrho'}$.

- If $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) = \emptyset$, then (i) must hold for all concepts B_β above (except in the cases where we have directly defined $\mathcal{B}_{S(y,z)} := \{\exists S_2\}$), and we define $\mathcal{B}$ as the union of all $\mathcal{B}_\beta$. In particular, we obtain that $c^j_{\varrho R} \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$. To show the two remaining conditions of Definition 3.7, we start again with the concept atoms $A(y) \in \psi$ with $\mathsf{f}(y) = \epsilon$. We know that

  $$\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \bigsqcap \mathcal{B}_{A(y)} \sqsubseteq B_{A(y)} = A,$$

  as required. Likewise, for any $(S_2, \mathcal{C}) \in (\mathsf{N_R^-} \times 2^{\mathsf{N_R^-}}) \cap \mathsf{range}(\mathsf{f})$ we know that there must be a role atom $S(y, z)$ such that $\mathsf{f}(z) = (S_2, \mathcal{C})$ and $\mathcal{O} \models S' \sqsubseteq S$ for some $S' \in \mathcal{C}$ since α is connected. But then we get

  $$\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \bigsqcap \mathcal{B}_{S(y,z)} \sqsubseteq B_{S(y,z)} = \exists S_2,$$

  as before.

  It remains to construct a tree witness query ψ as in Definition 4.6. In this construction, we maintain the invariant that $\mathcal{I}_j \models \psi$ via the homomorphism that maps all variables $y_{\varrho'}$ with $\varrho' \in \mathsf{range}(\mathsf{f})$ to $c^j_{\varrho R \varrho'|_1}$. We start by including in ψ all atoms from $\mathcal{B}|_\mathsf{R}(y_\epsilon)$, hence satisfying the first condition of Definition 4.6 and our invariant since we know that $c^j_{\varrho R} \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$. The second condition is immediately satisfied by the same arguments as above, by observing that, for any $A(y) \in \alpha$ with $\mathsf{f}(y) = \epsilon$, the set $\mathcal{B}_{A(y)} \subseteq \mathcal{B}$ contains only rigid basic concepts. Likewise, for all $(S_2, \mathcal{C}) \in \mathsf{range}(\mathsf{f})$ for which there is a role atom $S(y, z)$ with $\mathsf{f}(y) = \epsilon$ for which case (i) above applies, we know that $\mathcal{B}_{S(y,z)} \subseteq \mathcal{B}$ contains only rigid basic concepts whose conjunction implies $\exists S_2$. Hence, in this case $(S_2, \mathcal{C})$ is rigidly witnessed in ψ.

  However, if $(S_2, \mathcal{C}) \in \mathsf{range}(\mathsf{f})$ is such that S_2 is flexible and all role atoms of the form $S(y, z)$ for which $\mathsf{f}(y) = \epsilon$, $\mathcal{O} \models S' \sqsubseteq S$, and $S' \in \mathcal{C}$ are such that S' is rigid, then $\mathcal{B}$ contains only $\exists S_2$ and $\mathcal{C}$ consists of all these roles S'. If there is an alternative set $\mathcal{B}_{\exists S_2} \subseteq \mathsf{BC_R}(\mathcal{O})$ with $\mathcal{O} \models \bigsqcap \mathcal{B}_{\exists S_2} \sqsubseteq \exists S_2$ and $c^j_{\varrho R} \in (\bigsqcap \mathcal{B}_{\exists S_2})^{\mathcal{I}_j}$, then we can add $\mathcal{B}_{\exists S_2}(y_\epsilon)$ to ψ. Otherwise, we know that $(S_2, \mathcal{C})$ cannot be rigidly witnessed in ψ, and we have to add all atoms $S'(y_\epsilon, y_{(S_2,\mathcal{C})})$ to ψ in order to fulfill the last condition for $(S_2, \mathcal{C})$. Since we have $\mathcal{O} \models S_2 \sqsubseteq S'$ for all these S', the above atoms can be mapped into $\mathcal{I}_j$ as required for our invariant. Since $(S_2, \mathcal{C})$ is not rigidly witnessed in ψ, we further need to consider the atoms of α that are mapped below $c^j_{\varrho R S_2}$. We continue the construction of ψ by induction on the structure of f, until all remaining paths are already rigidly witnessed in ψ or we have reached a leaf of the tree described by f.

Assume hence that we have already defined ψ up to a variable of the form $y_{\varrho'}$, but that some $\varrho' \cdot (S_2, \mathcal{C}) \in \mathsf{range}(\mathsf{f})$ is not rigidly witnessed. We consider first all concept atoms $A(y) \in \alpha$ with $\mathsf{f}(y) = \varrho' \cdot (S_2, \mathcal{C})$. Since $c^j_{\varrho R \varrho'|_1 S_2} \in A^{\mathcal{J}_i}$, by Lemma 4.13 we know that either (i') there is a set $\mathcal{B}_A \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $\mathcal{O} \models \bigsqcap \mathcal{B}_A \sqsubseteq A$ and $c^j_{\varrho R \varrho'|_1 S_2} \in (\bigsqcap \mathcal{B}_A)^{\mathcal{I}_j}$, or (ii') $c^j_{\varrho R \varrho'|_1 S_2} \in A^{\mathcal{I}_j}$ and $\mathcal{W}_\mathcal{O}(c^j_{\varrho R \varrho'|_1 S_2}) \neq \emptyset$. In case (i'), we add the atoms $\mathcal{B}_A(y_{\varrho' \cdot (S_2, \mathcal{C})})$ to ψ to satisfy the corresponding condition of Definition 4.6, while maintaining our invariant. In case (ii'), there must be a prefix $\varrho'' R'$ of $\varrho R \varrho'|_1 S_2$ and a set $\mathcal{B}_{R'} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ such that $\mathcal{O} \models \bigsqcap \mathcal{B}_{R'} \sqsubseteq \exists R'$ and either $c^j_{\varrho''} \in (\bigsqcap \mathcal{B}_{R'})^{\mathcal{I}_j}$ or $\varrho'' \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$ and $\varrho'' \in (\bigsqcap \mathcal{B}_{R'})^{\mathcal{I}_j}$. If $\varrho'' R'$ is a prefix of ϱR, then this contradicts the fact that $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) = \emptyset$. But if ϱR is a prefix of $\varrho'' R'$, we would have added $\mathcal{B}_{R'}(y_{\varrho'''})$ to ψ earlier, where ϱ''' is the path in $\mathsf{range}(\mathsf{f})$ corresponding to ϱ'', and hence $\varrho' \cdot (S_2, \mathcal{C})$ would be rigidly witnessed in ψ. This shows that case (ii') is impossible.

Consider now any successor $\varrho' \cdot (S_2, \mathcal{C}) \cdot (S_3, \mathcal{C}') \in \mathsf{range}(\mathsf{f})$ and all role atoms $S(y, z) \in \alpha$ with $\mathsf{f}(y) = \varrho' \cdot (S_2, \mathcal{C})$, $\mathsf{f}(z) = \varrho' \cdot (S_2, \mathcal{C}) \cdot (S_3, \mathcal{C}')$, and $\mathcal{O} \models S' \sqsubseteq S$ for some $S' \in \mathcal{C}$. If S_3 is rigid or there is one such atom where S' is flexible (and hence $S' = S$), then we obtain as above that $c^j_{\varrho R \varrho'|_1 S_2} \in (\exists S_3)^{\mathcal{J}_i}$. By Lemma 4.13 and the same arguments as above, we know that there is a set $\mathcal{B}_{\exists S_3} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $\mathcal{O} \models \bigsqcap \mathcal{B}_{\exists S_3} \sqsubseteq \exists S_3$ and $c^j_{\varrho R \varrho'|_1 S_2} \in (\bigsqcap \mathcal{B}_{\exists S_3})^{\mathcal{I}_j}$. Hence, we can add the atoms $\mathcal{B}_{\exists S_3}(y_{\varrho' \cdot (S_2, \mathcal{C})})$ to ψ in order to ensure that $\varrho' \cdot (S_2, \mathcal{C}) \cdot (S_3, \mathcal{C}')$ is rigidly witnessed in ψ. If S_3 is flexible and there is no such atom and no set $\mathcal{B}_{\exists S_3}$ as above, then we again have to add all the atoms $S'(y_{\varrho' \cdot (S_2, \mathcal{C})}, y_{\varrho' \cdot (S_2, \mathcal{C}) \cdot (S_3, \mathcal{C}')})$ to ψ and continue the construction with $\varrho' \cdot (S_2, \mathcal{C}) \cdot (S_3, \mathcal{C}')$, which cannot be rigidly witnessed.

The construction of ψ terminates since f is finite, and when it does we have added enough rigid atoms to ψ in order to satisfy Definition 4.6, and furthermore know that $\mathcal{I}_j \models \psi$.

In both cases, we have found a witness query ψ for α such that $\mathcal{I}_j \models \psi$, and thus we obtain a contradiction from (R5) and Proposition 3.5 (recall that $\alpha \in Q_\mathsf{R}^\neg$).

(II) In the remainder of the proof, let π be such that it maps at least one term into $\Delta_\mathsf{n} := \mathsf{N_I}(\mathcal{K}) \cup \bigcup_{j=0}^{n+k} \Delta_\mathsf{a}^{\mathcal{I}_j} \cup \Delta_\mathsf{t}^{\mathcal{I}_j}$. In this case, we directly define a homomorphism π' of α into $\mathcal{I}_i$ in order to obtain a contradiction to the fact that $\mathcal{I}_i \models \neg\alpha$. We start defining π' for all terms $t \in \mathsf{N_V}(\alpha) \cup \mathsf{N_I}(\alpha)$ for which we have

$\pi(t) \in \Delta_{\mathsf{n}}$: if $\pi(t) = e^j$ for $e \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}} \cup \mathsf{N_I^{tree}}$, then we set $\pi'(t) := e^i$. We first show that

$$\text{for all } B \in \mathsf{BC}(\mathcal{O}) \text{ and all } t \in \mathsf{N_V}(\alpha) \cup \mathsf{N_I}(\alpha) \text{ with } \pi(t) \in \Delta_{\mathsf{n}}, \\ \text{we have } \pi'(t) \in B^{\mathcal{I}_i} \text{ whenever } \pi(t) \in B^{\mathcal{J}_i}. \quad (1)$$

By Lemma 4.13, $\pi(t) \in B^{\mathcal{J}_i}$ implies that (i) $\pi(t) \in B^{\mathcal{I}_i}$, or (ii) $\pi(t) \in \Delta_{\mathsf{a}}^{\mathcal{I}_j} \cup \Delta_{\mathsf{t}}^{\mathcal{I}_j}$ and there is a set $\mathcal{B} \subseteq \mathsf{BC_R}(\mathcal{O})$ such that $\pi(t) \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ and $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq B$. In case (i), we have $\pi'(t) = \pi(t)$, and hence the claim holds. In case (ii), if $i = j$, the claim follows as in case (i); otherwise, we further distinguish the following two cases.

- If $\pi(t) \in \Delta_{\mathsf{a}}^{\mathcal{I}_j}$, then $\pi(t)$ must be of the form a_x^j. Let $\beta \in \mathcal{Q}_\phi$ be the (unique) CQ containing the variable x. By (R3), the existence of the element a_x^j implies that $\beta \in Q_{\mathsf{R}}$. Hence, the element of the form a_x^i must also exist, i.e., $\pi'(t) = a_x^i$ is well-defined. Since only $\mathcal{A}_{Q_{\mathsf{R}}}$, $\mathcal{A}_{Q_{\iota(j)}}$, and $\mathcal{A}_{R_{\mathsf{F}}}$ contain assertions about a_x, by Definition 3.1 and Proposition 3.3 we know that the elements of $\mathcal{B}$ are implied by the conjunction of
  - all elements of $\mathsf{BC}^-(a_x, \beta)$, and
  - all rigid concepts $\exists R$ for which there is $\exists S(a_x) \in R_{\mathsf{F}}$ with $\mathcal{O} \models S \sqsubseteq R$.

  But for all concepts of the latter form, (R6) implies that $\exists R(a_x)$ is already implied by some $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}} \cup \mathcal{A}_{Q_{\mathsf{R}}} \cup \mathcal{A}_{Q_{\iota(i')}} \cup \mathcal{A}_{i'} \rangle$, and hence must follow also from $\bigsqcap \mathsf{BC}^-(a_x, \beta)$. This shows that $\mathcal{O} \models \bigsqcap \mathsf{BC}^-(a_x, \beta) \sqsubseteq \bigsqcap \mathcal{B}$, and hence $B'(a_x) \in \mathcal{A}_{Q_{\mathsf{R}}}$ for all $B' \in \mathcal{B}$. Since $\mathcal{I}_i \models \mathcal{A}_{Q_{\mathsf{R}}}$ and $\mathcal{I}_i \models \mathcal{O}$, we obtain that $a_x^i \in (\bigsqcap \mathcal{B})^{\mathcal{I}_i} \subseteq B^{\mathcal{I}_i}$, as required.

- If $\pi(t) \in \Delta_{\mathsf{t}}^{\mathcal{I}_j}$, then $\pi(t)$ is of the form $a_{b\varrho}^j$. In this case, the element $a_{b\varrho}^i$ also exists since $\mathcal{A}_{R_{\mathsf{F}}}$ is the same for all time points. By Proposition 3.3 and the definition of $\mathcal{A}_{R_{\mathsf{F}}}$, $\pi(t) \in B^{\mathcal{I}_j}$ implies that B subsumes the conjunction of all rigid basic concepts satisfied by $c_{b\varrho}$ in some $\mathcal{I}_{\langle \mathcal{O}, \{\exists S(b)\} \rangle}$ where $\exists S(b) \in R_{\mathsf{F}}$. Another application of Proposition 3.3 yields that B is also satisfied by $\pi'(t) = a_{b\varrho}^i$ in $\mathcal{I}_i$.

This concludes the proof of (1).

In particular, it follows that all concept atoms in α that are of the form $A(t)$ with $\pi(t) \in \Delta_{\mathsf{n}}$ are satisfied by π' in $\mathcal{I}_i$. We proceed by showing that this is also the case for all role atoms in α involving only terms of the above form. Let hence $R(t, t') \in \alpha$ be such that $\pi(t), \pi(t') \in \Delta_{\mathsf{n}}$. If $\pi(t)$ and $\pi(t')$ are both contained in $\Delta^{\mathcal{I}_i}$, the claim follows immediately from Lemma 4.12 and the fact that $\pi'(t) = \pi(t)$ and $\pi'(t') = \pi(t')$. If this is not the case, then since in $\mathcal{J}_i$ there are no role connections between elements of different sets $\Delta_{\mathsf{a}}^{\mathcal{I}_j} \cup \Delta_{\mathsf{t}}^{\mathcal{I}_j}$, $\pi(t)$ and $\pi(t')$ must both belong to some $\mathsf{N_I}(\mathcal{K}) \cup \Delta_{\mathsf{a}}^{\mathcal{I}_j} \cup \Delta_{\mathsf{t}}^{\mathcal{I}_j}$ for a fixed $j \neq i$, and it remains to consider the following cases:

- R is rigid and at least one of $\pi(t)$ or $\pi(t')$ is contained in $\Delta_{\mathsf{a}}^{\mathcal{I}_j}$, but neither is contained in $\Delta_{\mathsf{t}}^{\mathcal{I}_j}$. Since $(\pi(t), \pi(t')) \in R^{\mathcal{J}_i}$, we know that $(\pi(t), \pi(t')) \in R^{\mathcal{I}_j}$. By Definition 3.1, there must be an assertion $S(\tau(t), \tau(t')) \in \mathcal{A}_{Q_{\mathsf{R}}} \cup \mathcal{A}_{Q_{\iota(j)}}$ such that $\mathcal{O} \models S \sqsubseteq R$, where $\tau(t) := e$ if $\pi(t) = e^j$. By Definition 4.4, we get $R(\tau(t), \tau(t')) \in \mathcal{A}_{Q_{\mathsf{R}}}$. Since $\mathcal{I}_i \models \mathcal{A}_{Q_{\mathsf{R}}}$, we conclude $(\pi'(t), \pi'(t')) \in R^{\mathcal{I}_i}$.

- R is rigid and at least one of $\pi(t)$ or $\pi(t')$ is contained in $\Delta_{\mathsf{t}}^{\mathcal{I}_j}$. We again have $(\pi(t), \pi(t')) \in R^{\mathcal{I}_j}$. By Definition 3.1 and the definition of $\mathcal{A}_{R_{\mathsf{F}}}$, we know that $R(\tau(t), \tau(t')) \in \mathcal{A}_{R_{\mathsf{F}}}$. Since $\mathcal{I}_i \models \mathcal{A}_{R_{\mathsf{F}}}$, we get $(\pi'(t), \pi'(t')) \in R^{\mathcal{I}_i}$.

- R is flexible. Since $\mathcal{W}_{\mathcal{O}}$ is not defined for elements of Δ_{n}, there must be a rigid role S such that $(\pi(t), \pi(t')) \in S^{\mathcal{I}_j}$ and $\mathcal{O} \models S \sqsubseteq R$. As in the two previous cases, it follows that $(\pi'(t), \pi'(t')) \in S^{\mathcal{I}_i}$. Since $\mathcal{I} \models \mathcal{O}$, we obtain $(\pi'(t), \pi'(t')) \in R^{\mathcal{I}_i}$.

It remains to define π' for the variables of α that are mapped by π into $\bigcup_{j=0}^{n+k} \Delta_{\mathsf{u}}^{\mathcal{I}_j}$. Consider any such variable that is mapped to $c^j_{eRR_1\dots R_m}$ for some individual name $e \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}} \cup \mathsf{N_I^{tree}}$. Since α is connected, there must be a variable y with $\pi(y) = c^j_{eR}$ and an atom $S(t, y) \in \alpha$ such that $\mathcal{O} \models R \sqsubseteq S$ and $\pi(t) = e^j$. Recall that for all such atoms we have $\pi'(t) = e^i$. To determine the value of $\pi'(y)$, we consider all atoms of the above form. If $i = j$, then by Lemmas 4.12 and 4.13 all these atoms can be satisfied by setting $\pi'(y) := \pi(y) = c^i_{eR}$. Otherwise, we distinguish the following two cases.

- If $\mathcal{W}_{\mathcal{O}}(c^j_{eR}) \neq \emptyset$, then it follows from $(e^j, c^j_{eR}) \in R^{\mathcal{I}_j}$ that $(e^j, c^j_{eR}) \in R^{\mathcal{J}_i}$. By (1), we infer that $e^i = \pi'(t) \in (\exists R)^{\mathcal{I}_i}$, and hence the element c^i_{eR} also exists and the pair (e^i, c^i_{eR}) satisfies all role atoms $S(t, y)$ that are mapped to (e^j, c^j_{eR}) by π. Hence, we can set $\pi'(y) := c^i_{eR}$ for all such variables y.

- If $\mathcal{W}_{\mathcal{O}}(c^j_{eR}) = \emptyset$, then by the definition of $\mathcal{J}_i$ we know that for all atoms $S(t, y)$ as above there is a rigid role S' such that $\mathcal{O} \models R \sqsubseteq S' \sqsubseteq S$ and $(e^j, c^j_{eR}) \in (S')^{\mathcal{J}_i}$. Note that R must be flexible since otherwise $\exists R$ would be a witness for c^j_{eR}. In particular, we know that ϕ is not r-simple. Furthermore, since $(e^j, c^j_{eR}) \in R^{\mathcal{I}_j}$, Definition 3.1 and Proposition 3.3 yield that the assertion $\exists R(e)$ is implied by the basic concepts obtained from the assertions involving e in $\mathcal{K}^i_{\mathsf{R}}$. We now show by a case distinction on the form of e that this is already the case if we ignore the assertions in $\mathcal{A}_{R_{\mathsf{F}}}$.

  - If $e \in \mathsf{N_I}(\mathcal{K})$, then the basic concepts obtained from the assertions about e in $\mathcal{A}_{R_{\mathsf{F}}}$ are of the form $\exists R'$ for rigid roles R'. Since $\mathcal{A}_{\mathsf{R}}$ is an ABox type and $\mathcal{I}_j$ is a model of both $\mathcal{A}_{\mathsf{R}}$ and $\mathcal{A}_{R_{\mathsf{F}}}$, we must have $\exists R'(e) \in \mathcal{A}_{\mathsf{R}}$, and hence the assertions about e in $\mathcal{A}_{R_{\mathsf{F}}}$ are subsumed by $\mathcal{A}_{\mathsf{R}}$. Hence, $\exists R(e)$ follows from $\mathcal{K}^i_{\mathsf{R}}$ already without $\mathcal{A}_{R_{\mathsf{F}}}$.

– If $e = a_x \in \mathsf{N}_\mathsf{I}^\mathsf{aux}$ and β is the CQ in which x appears, then we know that $\exists R$ is implied by the conjunction of all elements of $\mathsf{BC}^-(a_x, \beta)$ and all rigid concepts $\exists R'$ for which there is a $\exists R''(a_x) \in R_\mathsf{F}$ with $\mathcal{O} \models R'' \sqsubseteq R'$. By (R6) and Proposition 3.3, all concepts of the latter form are implied by $\bigsqcap \mathsf{BC}^-(a_x, \beta)$, and hence $\exists R'(a_x)$ must be contained in $\mathcal{A}_{Q_\mathsf{R}}$ (see Definition 4.4). This shows again that $\exists R(e)$ follows from $\mathcal{K}^i_\mathsf{R}$ without $\mathcal{A}_{R_\mathsf{F}}$.
– If $e \in \mathsf{N}_\mathsf{I}^\mathsf{tree}$, then $\exists R(e)$ must follow exclusively from $\mathcal{A}_{R_\mathsf{F}}$ (and $\mathcal{O}$). Since $\mathcal{A}_{R_\mathsf{F}}$ contains only rigid assertions, the corresponding rigid basic concepts thus constitute a witness for c^j_{eR}, which contradicts our assumption.

We have thus shown the entailment required to apply (R6), and infer that $\exists R(e) \in \mathcal{A}_{R_\mathsf{F}}$. Since $\mathcal{I}_i \models \mathcal{A}_{R_\mathsf{F}}$, this means that $(e^i, a^i_{eR}) \in (S')^{\mathcal{I}_i}$ holds for all rigid roles S' identified above. This shows tat (e^i, a^i_{eR}) satisfies all role atoms $S(t, y)$ that are mapped to (e^j, c^j_{eR}) by π. We can thus define $\pi'(y) := a^i_{eR}$.

We have thus defined π' for all variables that are directly connected to some term t with $\pi(t) \in \Delta_\mathsf{n}$. We proceed to define π' by induction on the tree structure of the homomorphism π into $\mathcal{J}_i$ below these variables. It is easy to see that we do not have to change π on the variables y with $\pi(y) \in \Delta_\mathsf{u}^{\mathcal{I}_i}$; hence, in the following we consider only the case that $\pi(y) \in \Delta_\mathsf{u}^{\mathcal{I}_j}$ with $j \neq i$. In the construction of π', we maintain the invariant that whenever $\pi(y) = c^j_{\varrho R}$, then either (i) $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) = \emptyset$ and $\pi'(y) = a^i_{\varrho R}$ or (ii) $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) \neq \emptyset$ and $\pi'(y)$ is of the form $c^i_{\varrho' R}$.

Hence, assume that $\pi(y) = c^j_{\varrho R} \in \Delta_\mathsf{u}^{\mathcal{I}_j}$ and $\pi'(y)$ has already been defined. We first show that all concept atoms $A(y) \in \alpha$ are satisfied by π'. Since $c^j_{\varrho R} \in A^{\mathcal{J}_i}$, we know by Lemma 4.13 that either (i') there is a $\mathcal{B} \subseteq \mathsf{BC}_\mathsf{R}(\mathcal{O})$ with $c^j_{\varrho R} \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$ and $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq A$; or (ii') $c^j_{\varrho R} \in A^{\mathcal{I}_j}$ and $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) \neq \emptyset$. If case (ii) from above applies, i.e., we have $\pi'(y) = c^i_{\varrho' R}$, then two applications of Proposition 3.4 yield that $\mathcal{O} \models \exists R^- \sqsubseteq A$ and $\pi'(y) = c^i_{\varrho' R} \in A^{\mathcal{I}_i}$. If case (i) applies, then (i') must hold, and we know by Proposition 3.4 that $\mathcal{O} \models \exists R^- \sqsubseteq \bigsqcap \mathcal{B}$, and hence $\pi'(y) = a^i_{\varrho R} \in (\bigsqcap \mathcal{B})^{\mathcal{I}_i}$ by the definition of $\mathcal{A}_{R_\mathsf{F}}$. Since $\mathcal{I}_i \models \mathcal{O}$, we conclude that $\pi'(y) \in A^{\mathcal{I}_i}$.

To continue the definition of π', consider a fixed $\pi'(y')$ that has already been defined based on $\pi(y') = c^j_{\varrho R}$, and a fixed $c^j_{\varrho R S_2} \in \mathsf{range}(\pi)$. For all role atoms of the form $S(y, z)$ with $\pi(y) = c^j_{\varrho R}$ and $\pi(z) = c^j_{\varrho R S_2}$, we must have $\mathcal{O} \models S_2 \sqsubseteq S$. We again make a case distinction based on $\mathcal{W}_\mathcal{O}(c^j_{\varrho R S_2})$.

- If $\mathcal{W}_\mathcal{O}(c^j_{\varrho R S_2}) \neq \emptyset$, then we have $(c^j_{\varrho R}, c^j_{\varrho R S_2}) \in S_2^{\mathcal{J}_i}$ since this pair is contained in $S_2^{\mathcal{I}_j}$, and furthermore $\mathcal{O} \models \exists R^- \sqsubseteq \exists S_2$. One of the following two cases must apply:

– If $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) \neq \emptyset$, then we know that $\pi'(y) = c^i_{\varrho' R}$. The element $c^i_{\varrho' RS_2}$ must exist and $(c^i_{\varrho' R}, c^i_{\varrho' RS_2})$ satisfies all role atoms $S(y,z)$ of the above form in $\mathcal{I}_i$. Hence, we can define $\pi'(z) := c^i_{\varrho' RS_2}$ for all such variables z while maintaining the invariant (case (ii)).

– If $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) = \emptyset$, then there must be a set $\mathcal{B} \subseteq \mathsf{BC_R}(\mathcal{O})$ such that $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists S_2$ and $c^j_{\varrho R} \in (\bigsqcap \mathcal{B})^{\mathcal{I}_j}$, which implies $\mathcal{O} \models \exists R^- \sqsubseteq \bigsqcap \mathcal{B}$ by Proposition 3.4. Furthermore, $\pi'(y)$ must be of the form $a^i_{\varrho R}$, and hence $\pi'(y) \in (\bigsqcap \mathcal{B})^{\mathcal{I}_i}$ by the definition of $\mathcal{A}_{R_\mathsf{F}}$. Since $\mathcal{I}_i \models \mathcal{O}$, this implies that $\pi'(y) \in (\exists S_2)^{\mathcal{I}_i}$. Hence, the element $c^i_{a_{\varrho R} S_2}$ must exist and $(a^i_{\varrho R}, c^i_{a_{\varrho R} S_2})$ satisfies all required role atoms in $\mathcal{I}_i$, and we can set $\pi'(z) := c^i_{a_{\varrho R} S_2}$. This again satisfies case (ii) of our invariant.

- If $\mathcal{W}_\mathcal{O}(c^j_{\varrho RS_2}) = \emptyset$, then we know that also $\mathcal{W}_\mathcal{O}(c^j_{\varrho R}) = \emptyset$, and hence case (i) of our invariant applies, i.e., $\pi'(y) = a^i_{\varrho R}$. By the definition of $\mathcal{J}_i$, for every role atom $S(y,z)$ considered above there must be a rigid role S' such that $\mathcal{O} \models S' \sqsubseteq S$ and $(c^j_{\varrho R}, c^j_{\varrho RS_2}) \in (S')^{\mathcal{J}_i}$, and hence in particular $\mathcal{O} \models S_2 \sqsubseteq S'$. Furthermore, we know that the element $a^i_{\varrho RS_2}$ must exist since α is connected and π maps the terms of α into at least one element of Δ_n, and thus the length of ϱRS_2 cannot exceed the number of variables of α. Hence, by the definition of $\mathcal{A}_{R_\mathsf{F}}$ we know that $(a_{\varrho R}, a_{\varrho RS_2})$ satisfies all above roles S' in $\mathcal{I}_i$, and hence all relevant role atoms $S(y,z)$. We set $\pi'(z) := a_{\varrho RS_2}$ for all such variables, and obtain again case (i) of our invariant.

This concludes the construction of π' and shows that it is a homomorphism of α into $\mathcal{I}_i$, which contradicts our assumption that $\mathcal{I}_i \models \neg\alpha$. □

This finishes also the proof of Lemma 4.9. In the following sections, we use this characterization of r-satisfiability for obtaining different complexity bounds.

5 Regarding Combined Complexity

We now apply the procedure proposed in the previous section to show that—even with rigid symbols and TCQs that are not r-simple—the combined complexity of PSpace carries over from propositional LTL. For the latter, the satisfiability problem is PSpace-complete [Pnu77].

Theorem 5.1. *TCQ entailment in DL-Lite$^\mathcal{H}_{horn}$ is in* PSpace *w.r.t. combined complexity, even if* $\mathsf{N_{RR}} \neq \emptyset$.

The key insight of the previous section is that we do not need to store the exponentially large set $\mathcal{S}$ in order to check the conditions of Definition 4.8. It suffices to guess a polynomial-sized tuple $(\mathcal{A}_\mathsf{R}, Q_\mathsf{R}, Q_\mathsf{R}^\neg, R_\mathsf{F})$ in advance, and then check, in each step of an LTL-satisfiability test for ϕ^{p}, if there is a world $X_i \subseteq \{p_1, \ldots, p_m\}$ that satisfies the requirements specified in Definition 4.8.

For this purpose, we use the polynomial-space-bounded Turing machines for LTL-satisfiability constructed in [SC85]. Given the propositional LTL-formula ϕ^{p}, the machine $\mathcal{M}_{\phi^{\mathsf{p}}}$ iteratively guesses complete sets of (negated) subformulas of ϕ^{p} specifying which subformulas are satisfied at each point in time. Every such set induces a unique world $X_i \subseteq \{p_1, \ldots, p_m\}$ containing the propositional variables that are true.

In [SC85, Theorem 4.7], it is shown that if ϕ^{p} is satisfiable, then there must be a *periodic* model of ϕ^{p} with a period that is exponential in the size of ϕ^{p}. Hence, $\mathcal{M}_{\phi^{\mathsf{p}}}$ first guesses two polynomial-sized indices specifying the beginning and end of the first period. Then it continuously increments a (polynomial-sized) counter and in each step guesses a complete set of (negated) subformulas of ϕ^{p}. It then checks Boolean consistency of this set and consistency with the set of the previous time point according to the temporal operators. For example, if the previous set contains the formula $p_1 \mathsf{U}\, p_2$, then either it also contains p_2 or it must contain p_1 and the current set must contain $p_1 \mathsf{U}\, p_2$. In this way, the satisfaction of the U-formula is deferred to the next time point.

In each step, the oldest set is discarded and replaced by the next one. When the counter reaches the beginning of the period, it stores the current set and continues until it reaches the end of the period. At that point, instead of guessing the next set of subformulas, the set stored at the beginning of the period is used and checked for consistency with the previous set as described above. $\mathcal{M}_{\phi^{\mathsf{p}}}$ additionally has to ensure that all U-subformulas are satisfied within the period. Thus, the Turing machine never has to remember more than three sets of polynomial size.

Note that [SC85] do not directly regard past operators, which are considered by us. However, we can certainly adapt the complete sets of subformulas guessed by $\mathcal{M}_{\phi^{\mathsf{p}}}$ to also include the past operators. This does not affect the space requirements of the Turing machines; in particular, the period that has to be guessed is still exponential in the size of ϕ^{p}. We now modify this procedure to prove the desired PSPACE upper bound.

Let $\mathcal{K}$ be a TKB and ϕ be a TCQ. We analyze the complexity of the satisfiability problem by showing how $\mathcal{S}$ and ι satisfying the conditions of Lemma 4.3 can be found. For proving r-satisfiability of $\mathcal{S}$, it suffices to find a tuple $(\mathcal{A}_{\mathsf{R}}, Q_{\mathsf{R}}, Q_{\mathsf{R}}^{\neg}, R_{\mathsf{F}})$ satisfying conditions (R1)–(R6), by Lemma 4.9. All these conditions are such that it is not necessary to actually construct the whole set $\mathcal{S}$—it is enough to show that each world X_i we encounter when checking ϕ^{p} for satisfiability induces a knowledge base $\mathcal{K}_{\mathsf{R}}^i$ that satisfies all requirements.

We can thus run a modified version of the Turing machine $\mathcal{M}_{\phi^{\mathsf{p}}}$ that first guesses the sets $\mathcal{A}_{\mathsf{R}}$, Q_{R}, $Q_{\mathsf{R}}^{\neg}$, and R_{F} required by Definition 4.8, which can clearly be done in polynomial space, and then proceeds as before, but additionally executes the following checks for the world X induced by each guessed complete set of propositional subformulas:

(R1) Check the knowledge base $\mathcal{K}_\mathsf{R} = \langle \mathcal{O}, \mathcal{A}_\mathsf{R} \cup \mathcal{A}_{Q_\mathsf{R}} \cup \mathcal{A}_{Q_X} \cup \mathcal{A}_{R_\mathsf{F}} \cup \mathcal{A}_i \rangle$ for consistency.

The ABox $\mathcal{A}_i$ is only relevant for the first $n+1$ time points, after which it is empty,[7] and $\mathcal{A}_{Q_X}$ is formed by instantiating all CQs α_j where $p_j \in X$. The number of sets $\mathcal{C}_\mathcal{O}(\alpha_j)$ to be computed for $\mathcal{A}_{Q_\mathsf{R}}$ is equal to the cardinality of Q_R, which depends linearly on the size of ϕ. Hence, according to [ACKZ09], we need a number of P tests to compute each set $\mathcal{C}_\mathcal{O}(\alpha_j)$; this number depends linearly on the size of α_j (and hence that of ϕ) and polynomially on the size of the ontology $\mathcal{O}$.

Note that the exponentially large ABox $\mathcal{A}_{R_\mathsf{F}}$ can be ignored for this consistency test since once we have verified (R6), we know that for every $\mathcal{A}_{\exists S(b)} \subseteq \mathcal{A}_{R_\mathsf{F}}$ there is at least one world X' at which the existence of the elements described in $\mathcal{A}_{\exists S(b)}$ follows from the KB $\langle \mathcal{O}, \mathcal{A}_\mathsf{R} \cup \mathcal{A}_{Q_\mathsf{R}} \cup \mathcal{A}_{Q_{X'}} \cup \mathcal{A}_{X'} \rangle$, where $\mathcal{A}_{X'}$ is the ABox from $\mathcal{A}_0, \dots, \mathcal{A}_n$ associated to X' (or $\emptyset$ for any time point after n). Hence, for the rigid R-successors of b enforced by $\mathcal{A}_{\exists S(b)}$ we know that $\mathcal{A}_\mathsf{R}$ already contains $\exists R(b)$, and it is safe to add the remaining individuals from $\mathcal{A}_{\exists S(b)}$ to any model without causing a conflict with the other ABoxes or the ontology.

This consistency test can thus be done in P in the (polynomial) size of $\mathcal{K}_\mathsf{R}$ [ACKZ09] and thus needs only polynomial space.

(R2) Check, for each $p_j \in \overline{X}$, whether $\mathcal{K}_\mathsf{R} \not\models \alpha_j$ holds.

Using the non-deterministic version of the algorithm in [BAC10] (cf. the part after Theorem 12), this can be done by rewriting α_j using $\mathcal{O}$, nondeterministically choosing a CQ ψ from the resulting UCQ, and checking whether the interpretation $\mathcal{I}$ obtained from the ABox $\mathcal{A}_\mathsf{R} \cup \mathcal{A}_{Q_\mathsf{R}} \cup \mathcal{A}_{Q_X} \cup \mathcal{A}_{R_\mathsf{F}} \cup \mathcal{A}_i$ under the closed-world assumption satisfies ψ.

Since our ABox is of exponential size, but ψ is of size polynomial in the size of α_j and $\mathcal{O}$, it suffices to guess a polynomial part of the forest-shaped ABox $\mathcal{A}_{R_\mathsf{F}}$ in order to check whether there exists a homomorphism of ψ into $\mathcal{I}$, which can clearly be done in polynomial space.

(R3) Check, for each $p_j \in X$, whether $\alpha_j \in Q_\mathsf{R}$.

(R4) Check, for each $p_j \in \overline{X}$, whether $\alpha_j \in Q_\mathsf{R}^\neg$.

[7]For this reason, we also have to ensure that the guessed cycle does not start during the first $n+1$ time points, as the result of [SC85] only applies when there are no external conditions on the propositional models.

(R5) Check, for each tree-shaped $\alpha \in Q_{\mathsf{R}}^{-}$ and every witness query ψ of α w.r.t. $\mathcal{O}$, whether it holds that $\mathcal{K}_{\mathsf{R}} \not\models \psi$.

Since each ψ contains at most as many variables as α and at most quadratically many atoms over these variables, this non-entailment test can obviously be done in polynomial space by the same arguments as above. It remains to show how to enumerate all possible witness queries in PSpace.

The first step is to find all possible tree witnesses f for α. But since α is connected and f mimics the structure of α, we can easily enumerate all candidate mappings $\mathsf{f}\colon \mathsf{N_V}(\alpha) \to (\mathsf{N_R^-} \times 2^{\mathsf{N_R^-}})^*$ whose range consists only of sequences of polynomial size. Furthermore, the conditions of Definition 3.7 can be checked in polynomial time [ACKZ09]. By the same argument, we can enumerate all polynomial-sized sets $\mathcal{B} \subseteq \mathsf{BC}(\mathcal{O})$ and check whether they satisfy the conditions of $\mathsf{Con}(\alpha, \mathsf{f})$.

Based on f and $\mathcal{B}$, the construction of all possible tree witness queries can also be done in polynomial time and space (see Definition 4.6).

Finally, for the second option of Definition 4.7 we can construct the following graph over all roles of $\mathsf{N_R^-}(\mathcal{O})$: it contains an edge from R_1 to R_2 iff $\mathcal{O} \models \exists R_1^- \sqsubseteq \exists R_2$. Again, this graph can be constructed by quadratically many P-tests [ACKZ09]. It then suffices to test whether it is possible to reach an R_2 satisfying $\{R_2\} \in \mathsf{Con}(\alpha, \mathsf{f})$ from an R_1 with $\mathcal{O} \models \bigsqcap \mathcal{B} \sqsubseteq \exists R_1$ for some $\mathcal{B} \subseteq \mathsf{BC_R}(\mathcal{O})$. All this is clearly possible within polynomial space.

(R6) Check, for each $S \in \mathsf{N_R}(\mathcal{O}) \setminus \mathsf{N_{RR}}$ and $b \in \mathsf{N_I}(\mathcal{K}) \cup \mathsf{N_I^{aux}}$, whether from $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}} \cup \mathcal{A}_{Q_{\mathsf{R}}} \cup \mathcal{A}_{Q_X} \cup \mathcal{A}_i \rangle \models \exists S(b)$ it follows that $\exists S(b) \in R_{\mathsf{F}}$.

This can be done in polynomial space by the same arguments as above.

Note that the other direction of Condition (R6) cannot be checked locally. Instead, it requires us to maintain for every $\exists S(b) \in R_{\mathsf{F}}$ the additional information whether we have already encountered a world where the above test has succeeded. This is also possible in polynomial space, but may require us to look for an LTL-structure with a longer period. However, since this condition basically behaves like linearly many additional temporal $\Diamond$-formulas that have to be checked, the maximal required period is still exponential in the input and can be represented in polynomial space.

The set $\mathcal{S}$ required for Lemma 4.3 corresponds to the set of all worlds X encountered during a run of this modified Turing machine, while ι can be obtained collecting the worlds guessed for the first $n+1$ time points. Under these definitions of $\mathcal{S}$ and ι, it is easy to see that the above checks are actually equivalent to (R1)–(R6) from Definition 4.8. By Lemmas 4.3 and 4.9, the described Turing machine accepts the input $\mathcal{K}$ and ϕ iff ϕ has a model w.r.t. $\mathcal{K}$. Since we do not have to store $\mathcal{S}$ explicitly and all checks can be done with a nondeterministic Turing machine using only polynomial space, according to [Sav70], TCQ entailment can be decided in PSpace. This finishes the proof of Theorem 5.1.

6 Regarding Data Complexity

In this section, we show that the low data complexity of query answering in *DL-Lite* does not increase dramatically in the temporal setting if we stay within—for which it is in ALogTime. We establish the corresponding hardness for *DL-Lite*$_{core}$ and thus prove that TCQ entailment is not FO-rewritable in any DL of the *DL-Lite* family.

Theorem 6.1. *TCQ entailment in DL-Lite*$_{core}$ *is* ALogTime*-hard w.r.t. data complexity, even if* $\mathsf{N_{RC}} = \emptyset$ *and* $\mathsf{N_{RR}} = \emptyset$*.*

Proof. There are regular languages that are NC^1-complete w.r.t. constant-depth reductions [BCST92, Theorem 7], and for any regular language there is an NFA recognizing it. Furthermore, the complexity class of DLogTime-uniform NC^1 equals that of ALogTime [MBIS90, Lemma 7.2].

We hence can establish ALogTime-hardness by considering an arbitrary NFA $\mathfrak{A}$ and reducing its word problem to TCQ entailment. Note that we adapt a construction of [AKK$^+$14, AKK$^+$15]. We consider concept names A_a and Q_q for characters a of the input alphabet and states q, respectively, and define the TCQ

$$\phi := \Box^- \Big(\bigwedge_{q \to_a q'} \big(Q_q(a) \wedge A_a(a) \big) \to \bigcirc Q_{q'}(a) \Big) \to Q_{q_1}(a),$$

where q_1 is the accepting state of $\mathfrak{A}$. Given an input word $w = a_0 \dots a_{n-1}$, we define the sequence of ABoxes $\mathcal{A}_w = (\mathcal{A}_i)_{0 \leq i < n}$ such that $\mathcal{A}_0 := \{Q_{q_0}(a)\}$ and $\mathcal{A}_i := \{A_{a_i}(a)\}$, for all $0 \leq i < n$, with q_0 being the initial state of $\mathfrak{A}$. Thus, $\mathfrak{A}$ accepts w iff all models of $\langle \emptyset, \mathcal{A}_w \rangle$ that satisfy the antecedent of ϕ (which means that they simulate all runs of $\mathfrak{A}$ on w), also satisfy the consequent $Q_{q_1}(a)$, which is equivalent to the entailment $\langle \emptyset, \mathcal{A}_w \rangle \models \phi$. □

In the remaining parts of this section, we show how the corresponding upper bound can be obtained. As a first step, we show that r-satisfiability is FO-rewritable. Based on that we subsequently present a procedure to solve the TCQ entailment problem in ALogTime w.r.t. data complexity.

6.1 A FO Rewriting for r-satisfiability

In this section, we use the notion of r-completeness (cf. Definition 4.8) to construct a set of first-order formulas which can be used to decide if a set $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$ is r-satisfiable w.r.t. a mapping ι and a TKB $\langle \mathcal{O}, (\mathcal{A}_i)_{0 \leq i \leq n} \rangle$; by evaluating the formulas over the sequence $(\mathcal{A}_i)_{0 \leq i \leq n}$, considered as one finite first-order structure DB.

We start specifying the two-sorted structure DB more formally, over the individual domain $\mathsf{N_I}(\mathcal{K})$ and temporal domain $\mathsf{N_T}(\mathcal{K}) := \{-1, 0, \dots, n\}$. We define the below relations, for all $B \in \mathsf{BC}(\mathcal{O})$ and $R \in \mathsf{N_R^-}(\mathcal{O})$:

$$\mathsf{B}^{\mathsf{DB}} := \{(a, i) \mid i \in \{0, \dots, n\},\ B(a) \in \mathcal{A}_i\}$$
$$\mathsf{R}^{\mathsf{DB}} := \{(a, b, i) \mid i \in \{0, \dots, n\},\ R(a, b) \in \mathcal{A}_i\}.$$

The semantics of the satisfaction relation $\models$ defined in the usual way, i.e., we have $\mathsf{DB} \models \mathsf{B}(a, i)$ iff $(a, i) \in \mathsf{B}^{\mathsf{DB}}$, and $\mathsf{DB} \models \mathsf{R}(a, b, i)$ iff $(a, b, i) \in \mathsf{R}^{\mathsf{DB}}$. The temporal domain element -1 allows us to access the empty ABox via $\mathsf{B}(a, -1)$ and $\mathsf{R}(a, b, -1)$.

Since the TBox and the query are fix, we can define several abbreviations, which facilitate the definition of our rewriting. Subsequently, we show that the r-satisfiability part is FO-rewritable, and later use this in the definition of an ALogTime Turing machine for the TCQ entailment problem.

Recall that we need to find a set $\mathcal{S}$ and mapping ι for which both the r-satisfiability and the LTL-satisfiability test succeed. To test the r-satisfiability, we rely on the results obtained in [BAC10] for knowledge base consistency and (U)CQ entailment. These decision problems are solved in [BAC10] by evaluating Boolean UCQs (with inequalities)—called $q_{unsat(\mathcal{O})}$ and $\mathsf{PerfectRef}(\psi, \mathcal{O})$, respectively—which are independent of the ABox $\mathcal{A}$, over the FO-structure $DB(\mathcal{A})$, which represents a minimal model of $\mathcal{A}$ and is constructed independently of the ontology or the query. In particular, $DB(\mathcal{A})$ contains a relation for every concept and role name, and exactly one tuple for every assertion in the ABox $\mathcal{A}$ (see our definition of DB[8]).

We now adapt this construction for our purpose (i.e., to decide the problems described in Definition 4.8) in that we not only consider the ABoxes $\mathcal{A}_i$, but also include $\mathcal{A}_{\mathsf{R}}$, $\mathcal{A}_{Q_{\mathsf{R}}}$, $\mathcal{A}_{Q_j}$, and $\mathcal{A}_{R_{\mathsf{F}}}$ for deciding consistency and entailment of UCQs.

However, regarding data complexity, it would not be practical to consider the entire set R_{F} and ABox $\mathcal{A}_{R_{\mathsf{F}}}$ as defined in Definition 4.8 within our rewriting—recall that $\mathcal{A}_{R_{\mathsf{F}}}$ may contain auxiliary elements tailored to individual elements in $\mathsf{N_I}(\mathcal{K})$, which then would need to be considered explicitly within the rewriting. Therefore, we discern the assertions in R_{F} more fine-granularly, according to the kind of individual they address. More precisely, we consider the set R_{F} to be the disjoint union of the three sets $R_{\mathsf{F|aux}}$, $R_{\mathsf{F}|\phi}$, and $R_{\mathsf{F|o}}$ (o for 'other'), each containing only assertions on the individuals of $\mathsf{N_I^{aux}}$, $\mathsf{N_I}(\phi)$, and $\mathsf{N_I}(\mathcal{K}) \setminus \mathsf{N_I}(\phi)$, respectively. The corresponding ABoxes $\mathcal{A}_{R_{\mathsf{F|aux}}}$ etc. are then defined in correspondence to the definition of $\mathcal{A}_{R_{\mathsf{F}}}$ in Definition 4.8.

Further, note that the set $R_{\mathsf{F}|\phi}$ depends on the mapping ι. This is because it covers the individuals that occur within ϕ, and hence the ABoxes $\mathcal{A}_i$ and $\mathcal{A}_{Q_{X_{\iota(i)}}}$

[8]Note that we follow the approach of [CDGL+05] by introducing a relation for every basic concept, and not just for the concept names.

influence its shape. However, since it only covers a fix number of individuals, the size of this set, and the size of the corresponding ABox $\mathcal{A}_{R_{\mathsf{F}|\phi}}$, is constant w.r.t. data complexity.

We now define our 'auxiliary' ABoxes in dependence of two given sets of constant size, $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1,\ldots,p_m\}}$ and $\mathcal{B}_\phi \subseteq \{B(a) \mid B \in \mathsf{BC}(\mathcal{O}),\ a \in \mathsf{N_I}(\phi)\}$. Note that we already defined $\mathcal{A}_{Q_j}$, $1 \le j \le k$, in Section 4; for convenience, we sometimes write $\mathcal{A}_{Q_{X_j}}$ instead of $\mathcal{A}_{Q_j}$. In addition, we define the following sets:

$$\begin{aligned}
Q_{\mathsf{R}[\mathcal{S}]} &:= \{\alpha_j \in \mathcal{Q}_\phi \mid X \in \mathcal{S},\ p_j \in X\},\\
Q^{-}_{\mathsf{R}[\mathcal{S}]} &:= \{\alpha_j \in \mathcal{Q}_\phi \mid X \in \mathcal{S},\ p_j \notin X\}, \text{ and}\\
\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} &:= \bigcup_{\alpha \in Q_{\mathsf{R}[\mathcal{S}]}} \mathcal{C}_{\mathcal{O}}(\alpha).
\end{aligned}$$

The construction of the set $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ is however more involved. This is because we cannot assume a set $\mathcal{B}_{\mathsf{o}}$ (i.e., in correspondence with $\mathcal{B}_\phi$ but covering all other elements of $\mathsf{N_I}(\mathcal{K})$) to be given. For that reason, we have to model the derivation of all relevant (rigid) basic concept assertions that are entailed from the TKB in our rewriting. This is specified below, where we assume $\mathcal{B}_{\mathsf{R}|\phi}$ to denote the set of all rigid assertions in $\mathcal{B}_\phi$, and correspondingly for $\mathcal{B}_{\mathsf{R}|\mathsf{o}}$.

$$\begin{aligned}
\mathcal{B}^0_{\mathsf{R}|\mathsf{o}} &:= \emptyset,\\
\mathcal{B}^{j+1}_{\mathsf{R}|\mathsf{o}} &:= \{B(a) \mid B \in \mathsf{BC_R}(\mathcal{O}),\ a \in \mathsf{N_I}(\mathcal{K}) \setminus \mathsf{N_I}(\phi),\\
&\qquad \exists i. 0 \le i \le n,\ \langle \mathcal{O}, \mathcal{B}^{j}_{\mathsf{R}|\mathsf{o}} \cup \mathcal{A}_i \rangle \models B(a)\},\\
\mathcal{B}_{\mathsf{R}|\mathsf{o}} &:= \mathcal{B}^{|\mathsf{BC_R}(\mathcal{O})|}_{\mathsf{R}|\mathsf{o}},\\
R_{\mathsf{F}|\mathsf{aux}[\mathcal{S}]} &:= \{\exists S(a_y) \mid S \in \mathsf{N^-_R}(\mathcal{O}) \setminus \mathsf{N^-_{RR}},\ a_y \in \mathsf{N^{aux}_I},\\
&\qquad \exists X. X \in \mathcal{S},\ \langle \mathcal{O}, \mathcal{A}_{Q_X} \rangle \models \exists S(a_y)\},\\
R_{\mathsf{F}|\phi[\mathcal{B}_\phi]} &:= \{\exists S(a) \in \mathcal{B}_\phi \mid S \in \mathsf{N^-_R}(\mathcal{O}) \setminus \mathsf{N^-_{RR}}\},\\
R_{\mathsf{F}|\mathsf{o}} &:= \{\exists S(a) \mid S \in \mathsf{N^-_R}(\mathcal{O}) \setminus \mathsf{N^-_{RR}},\ a \in \mathsf{N_I}(\mathcal{K}) \setminus \mathsf{N_I}(\phi),\\
&\qquad \exists i. 0 \le i \le n,\ \langle \mathcal{O}, \mathcal{B}_{\mathsf{R}|\mathsf{o}} \cup \mathcal{A}_i \rangle \models \exists S(a)\},\\
\mathcal{A}^{+}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} &:= \mathcal{B}_{\mathsf{R}|\phi} \cup \mathcal{B}_{\mathsf{R}|\mathsf{o}} \cup \{\exists R(a) \mid R(a,e) \in \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}},\ a \in \mathsf{N_I}(\phi)\} \cup\\
&\qquad \{R(a,b) \mid R \in \mathsf{N_{RR}}(\mathcal{O}),\ a,b \in \mathsf{N_I}(\mathcal{K}),\ R(a,b) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \text{ or}\\
&\qquad \exists i. 0 \le i \le n, \langle \mathcal{O}, \mathcal{A}_i \rangle \models R(a,b)\}
\end{aligned}$$

$\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ is then defined as the union of $\mathcal{A}^{+}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ and all negative rigid role and basic concept assertions $\neg\alpha$ over $\mathsf{N_I}(\mathcal{K})$ for which we have $\alpha \notin \mathcal{A}^{+}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. Note also that the sets $R_{\mathsf{F}|\mathsf{aux}[\mathcal{S}]}$ and $R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$ are constant.

The last component of $\mathcal{A}^{+}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ is necessary to prove that all basic concept assertions about individuals in $\mathsf{N_I}(\phi)$ that are entailed by $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$ are already entailed by $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ (see Condition (R6)).

We next provide an auxiliary Lemma.

Lemma 6.2. *For all $B \in \mathsf{BC_R}(\mathcal{O}), R \in \mathsf{N_{RR}}(\mathcal{O})$ and $a, b \in \mathsf{N_I}(\mathcal{K}) \setminus \mathsf{N_I}(\phi)$, we have*

- *$B(a) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ iff there is an $i, 0 \leq i \leq n$, with $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i \rangle \models B(a)$; and*
- *$R(a,b) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ iff there is $i, 0 \leq i \leq n$, with $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i \rangle \models R(a,b)$.*

Proof. The direction ($\Rightarrow$) is trivial.

We consider ($\Leftarrow$) and assume that $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i \rangle \models B(a)$ holds for some i, $0 \leq i \leq n$. Observe that, if we have $\mathcal{B}^j_{\mathsf{R|o}} = \mathcal{B}^{j+1}_{\mathsf{R|o}}$ at some point, then we have $\mathcal{B}^j_{\mathsf{R|o}} = \mathcal{B}^{j+l}_{\mathsf{R|o}}$, for all $l \geq 0$. But then, there must be some $j', 0 \leq j' \leq |\mathsf{BC_R}(\mathcal{O})|$, such that $\mathcal{B}^{j'}_{\mathsf{R|o}} = \mathcal{B}^{j'+l}_{\mathsf{R|o}}$, for all $l \geq 0$. This is because every set $\mathcal{B}^{j+1}_{\mathsf{R|o}}$ for which we have $\mathcal{B}^{j+1}_{\mathsf{R|o}} \neq \mathcal{B}^j_{\mathsf{R|o}}$ must contain at least one new assertion, there are only $|\mathsf{BC_R}(\mathcal{O})|$ relevant assertions per individual (in $\mathsf{N_I}(\mathcal{K}) \setminus \mathsf{N_I}(\phi)$), and, by Proposition 3.3, an assertion on a specific individual does not depend on assertions on other individuals. We then get that $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i \rangle \models B(a)$ leads to $\langle \mathcal{O}, \mathcal{B}^{|\mathsf{BC_R}(\mathcal{O})|}_{\mathsf{R|o}} \cup \mathcal{A}_i \rangle \models B(a)$. This is because, by Proposition 3.3, only the assertions on the individual a (possibly contained in a tuple of individuals) in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ are relevant for the entailment; that is, in addition to the assertions in $\mathcal{B}_{\mathsf{R|o}}$, we have to consider those entailed by $\langle \mathcal{O}, \mathcal{A}_i \rangle$, for some $0 \leq i \leq n$; but the basic concept assertions corresponding to these role assertions have to be contained in $\mathcal{B}^1_{\mathsf{R|o}}$ and hence also in $\mathcal{B}_{\mathsf{R|o}}$, by definition. Hence $B(a) \in \mathcal{B}^{|\mathsf{BC_R}(\mathcal{O})|+1}_{\mathsf{R|o}}$. By our above observations, we thus can conclude that $B(a) \in \mathcal{B}^{|\mathsf{BC_R}(\mathcal{O})|}_{\mathsf{R|o}}$ and thus $B(a) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, by the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$.

We second assume $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i \rangle \models R(a,b)$. By Definition 3.1 and Proposition 3.5, we must have some $T \in \mathsf{N_R^-}$ such that $T(a,b) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i$ and $\mathcal{O} \models T \sqsubseteq R$. If $T(a,b) \in \mathcal{A}_i$, we have that $\langle \mathcal{O}, \mathcal{A}_i \rangle \models R(a,b)$, and hence $R(a,b) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ by the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. Otherwise, there must be a j, $0 \leq j \leq n$, such that $\langle \mathcal{O}, \mathcal{A}_j \rangle \models T(a,b)$, and hence also $\langle \mathcal{O}, \mathcal{A}_j \rangle \models R(a,b)$, which shows that $R(a,b) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ as above. □

Given the above definitions and a mapping $\iota\colon \{0, \ldots, n\} \to \{1, \ldots, k\}$, we regard the KBs

$$\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} = \langle \mathcal{O}, \mathcal{A}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}} \rangle,$$

for all i, $0 \leq i \leq n + k$, for query answering, where

$$\mathcal{A}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}} := \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_{R_{\mathsf{F}[\mathcal{S},\mathcal{B}_\phi]}} \cup \mathcal{A}_i$$

with $\mathcal{A}_i := \emptyset, n < i \leq n + k$, and $\mathcal{A}_{R_{\mathsf{F}[\mathcal{S},\mathcal{B}_\phi]}} := \mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}} \cup \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}} \cup \mathcal{A}_{R_{\mathsf{F|o}}}$. Observe that, to construct $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ and $R_{\mathsf{F|o}}$, we already need to decide entailment w.r.t. different KBs that contain our auxiliary ABoxes and where $i, 0 \leq i \leq n$, is considered as a parameter. Therefore, we will define first custom versions of PerfectRef before specifying the final rewriting, which is then based on $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ etc. First, we however propose an approach to deal with the data dependence of the ABox $\mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}$.

Dealing with $R_{\mathsf{F}|\mathsf{o}}$

Since the rewriting must not depend on the contents of the database, we need to take special care of $\mathcal{A}_{R_{\mathsf{F}}}$, particularly $\mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}$. We now propose an approach for dealing with that latter set.

Specifically, we consider prototypical elements of the form $[S], a_{[S]S}, a_{[S]S\varrho}$, etc., $S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O})$, for those occurring in $\mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}$ (i.e., $[S]$ is used instead of a concrete individual name). We collect all these elements in the set $\mathsf{N}_\mathsf{I}^{\mathsf{pro}}$ and, for each $S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O}) \setminus \mathsf{N}_{\mathsf{RR}}$, define the ABox $\mathcal{A}_{\exists S}$ as the a prototypical version of an ABox $\mathcal{A}_{\exists S(b)}, b \in \mathsf{N}_\mathsf{I}$, which is obtained from $\mathcal{A}_{\exists S(b)}$ by replacing b by $[S]$, which we assume to be a fresh individual name, in all assertions in $\mathcal{A}_{\exists S(b)}$.

For a CQ $\psi := \exists\vec{x}.\varphi$, we then define the rewriting $\psi^\dagger := \exists\vec{x}.\varphi \wedge \varphi_{\mathsf{filter}}$, with

$$\varphi_{\mathsf{filter}} := \bigwedge_{\substack{R\in\mathsf{N}_\mathsf{R}^-,\\ R(t_1,t_2)\in\psi}} \left(\neg\mathsf{pro}(t_1) \wedge \mathsf{pro}(t_2) \rightarrow \bigwedge_{\substack{S\in\mathsf{N}_\mathsf{R}^-,\\ S(t_2,t_3)\in\psi}} (t_1 = t_3 \vee \mathsf{pro}(t_3)) \right),$$

where we assume pro to be a unary predicate that identifies exactly the elements of $\mathsf{N}_\mathsf{I}^{\mathsf{pro}}$. Furthermore, given $\mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}$, the ABox $\mathcal{A}^\dagger_{R_{\mathsf{F}|\mathsf{o}}}$ is defined as the union of $\bigcup_{\exists S(a)\in\mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}} \mathcal{A}_{\exists S}$ and the set containing all assertions of the form $\mathsf{pro}(a)$, for all $a \in \mathsf{N}_\mathsf{I}^{\mathsf{pro}}$ occurring in the former set. The below lemma captures the intent of this rewriting.

Lemma 6.3. *Let $\psi := \exists\vec{x}.\varphi$ be a CQ and $\mathcal{A}\cup\mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}$ be one of the ABoxes $\mathcal{A}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}$, $0 \leq i \leq n+k$. Then, we have*

$$DB(\mathcal{A} \cup \mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}) \models \psi \textit{ iff } DB(\mathcal{A} \cup \mathcal{A}^\dagger_{R_{\mathsf{F}|\mathsf{o}}}) \models \psi^\dagger.$$

Proof. ($\Rightarrow$) Let π be a homomorphism of ψ into $DB(\mathcal{A} \cup \mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}})$. We define a homomorphism π' of $\psi^\dagger$ into $DB(\mathcal{A} \cup \mathcal{A}^\dagger_{R_{\mathsf{F}|\mathsf{o}}})$. More precisely, π' corresponds to π but maps elements from $\mathsf{N}_\mathsf{I}^{\mathsf{tree}}$ that do not occur in $\mathcal{A}^\dagger_{R_{\mathsf{F}|\mathsf{o}}}$ to the corresponding prototypes, which must exist in the domain of $DB(\mathcal{A} \cup \mathcal{A}^\dagger_{R_{\mathsf{F}|\mathsf{o}}})$, by the definition of $\mathcal{A}^\dagger_{R_{\mathsf{F}|\mathsf{o}}}$.

Let now $R(t_1,t_2), S(t_2,t_3) \in \psi, R, S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O})$ be arbitrary, $\pi'(t_1) \notin \mathsf{N}_\mathsf{I}^{\mathsf{pro}}$, and $\pi'(t_2) \in \mathsf{N}_\mathsf{I}^{\mathsf{pro}}$, such that the precondition of $\varphi_{\mathsf{filter}}$ evaluates to true under π'. We thus have $\pi'(t_1) = \pi(t_1)$ and $\pi(t_2) \in \mathsf{N}_\mathsf{I}^\mathsf{tree}$, by our definition of π'. Recall that $\mathcal{A}$ does not contain assertions on the element $\pi(t_2)$ of $\mathsf{N}_\mathsf{I}^\mathsf{tree}$ that we considered for our replacement; hence, the definition of DB implies that $R(\pi(t_1), \pi(t_2))$ and $S(\pi(t_2), \pi(t_3))$ are contained in $\mathcal{A}_{R_{\mathsf{F|o}}}$. Together with $\pi'(t_1) \notin \mathsf{N}_\mathsf{I}^\mathsf{pro}$ and $\pi'(t_1) = \pi(t_1)$, this means $\pi(t_1) \notin \mathsf{N}_\mathsf{I}^\mathsf{tree}$, by our definition of π'. Since $\pi(t_1)$ occurs in $\mathcal{A}_{R_{\mathsf{F|o}}}$, we hence have $\pi(t_1) \in \mathsf{N}_\mathsf{I}(\mathcal{K}) \setminus \mathsf{N}_\mathsf{I}(\phi)$. By the construction of $\mathcal{A}_{R_{\mathsf{F|o}}}$ and the two role atoms contained in it, we thus must have $\pi(t_3) = \pi(t_1)$ or $\pi(t_3) \in \mathsf{N}_\mathsf{I}^\mathsf{tree}$. This yields $\pi'(t_3) = \pi'(t_1)$ or $\pi'(t_3) \in \mathsf{N}_\mathsf{I}^\mathsf{pro}$, by our definition of π'. Hence, π' satisfies φ_filter.

($\Leftarrow$) Let π' be a homomorphism of $\psi^\dagger$ into $DB(\mathcal{A} \cup \mathcal{A}^\dagger_{R_{\mathsf{F|o}}})$. We construct a homomorphism π of ψ into $DB(\mathcal{A} \cup \mathcal{A}_{R_{\mathsf{F|o}}})$.

If π' does map to elements of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$, but not to elements of $\mathsf{N}_\mathsf{I}(\mathcal{K}) \setminus \mathsf{N}_\mathsf{I}(\phi)$, it may only map to elements of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$. This is because we assume ψ to be connected, and role atoms containing terms mapped to elements of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$ can only be satisfied by $DB(\mathcal{A} \cup \mathcal{A}^\dagger_{R_{\mathsf{F|o}}})$ through assertions in $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$, which only contains individuals of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$ and $\mathsf{N}_\mathsf{I}(\mathcal{K}) \setminus \mathsf{N}_\mathsf{I}(\phi)$. Further, note that the elements of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$ in $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$ are connected in tree structures, and that, for each such structure, there is at least one corresponding structure in $\mathcal{A}_{R_{\mathsf{F|o}}}$, by the definition of $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$. We hence can define π to map to the corresponding tree elements, in one such structure which corresponds to that π' maps to.

We consider the remaining case where π' maps to elements of both $\mathsf{N}_\mathsf{I}^\mathsf{pro}$ and $\mathsf{N}_\mathsf{I}(\mathcal{K}) \setminus \mathsf{N}_\mathsf{I}(\phi)$. We first define π as π' w.r.t. all elements other than those of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$. Note that $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$ does not contain assertions which do *not* contain elements of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$. By the definition of DB, we thus have $\alpha \in \mathcal{A}$, for all atoms $\alpha \in \psi$ where $\pi'(\alpha)$ does not contain elements of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$. We now assume t_1 to be a term in ψ that is mapped to an element of $\mathsf{N}_\mathsf{I}(\mathcal{K}) \setminus \mathsf{N}_\mathsf{I}(\phi)$ and occurs in a role atom $R(t_1, t_2)$ together with a term t_2 mapped to an element of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$; since we assume ψ to be connected, such a role atom must exist. Then, the definition of DB; that of $\mathcal{A}$, which does not contain elements of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$; and that of $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$, based on that of $\mathcal{A}_{R_{\mathsf{F|o}}}$, together yield that $\pi'(t_2) \in \mathsf{N}_\mathsf{I}^\mathsf{pro}$ is of the form $\pi'(t_2) = a_{[S]S}, S \in \mathsf{N}_\mathsf{R}^-$. We now define $\pi(t_2) = a_{\pi'(t_1)S}$. Note that, by the fact that the filter conjunct is satisfied, we have that all role atoms in which t_2 occurs only contain terms t_3 that are either mapped to $\pi'(t_1)$ or to some element of $\mathsf{N}_\mathsf{I}^\mathsf{pro}$. The former kind of role atoms is hence satisfied by this definition of π already, given the construction of $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$ from $\mathcal{A}_{R_{\mathsf{F|o}}}$. For the other kind of atoms, we set $\pi(t_3) = a_{\pi'(t_1)S\varrho}$, for $\pi'(t_3) = a_{[S]S\varrho}$. The proof can be continued by induction on the structure of π' below $\pi'(t_1)$, in the same way as above, and it is easy to verify that this definition of π is as required. □

If we apply this lemma in the following, we always assume $\mathsf{N}_\mathsf{I}^\mathsf{tree}$ to only contain the auxiliary elements used in $\mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}$ and $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$ (i.e., it contains neither those from $\mathsf{N_I^{pro}}$, nor the original auxiliary elements of $\mathcal{A}_{R_{\mathsf{F|o}}}$).

First Variants of PerfectRef

As mentioned above, the original definition of PerfectRef captures the ontological knowledge and targets an atemporal database as described in [CDL+07]. The goal of this section is to propose first variants of PerfectRef that similarly include the ontology, but target our time-stamped database DB and may also include some of our auxiliary ABoxes for answering a given CQ ψ, which we assume to contain only individual names from $\mathsf{N_I}(\phi)$.

As a first step, we construct the query $\mathsf{pr}_{(\psi,\mathcal{O})}(i)$ by simply replacing all atoms $B(t_1)$, $R(t_1,t_2)$ in $\mathsf{PerfectRef}(\psi,\mathcal{O})$ by $\mathsf{B}(t_1,i)$ and $\mathsf{R}(t_1,t_2,i)$, respectively, in the given CQ ψ. The adapted query, when asked over DB, thus decides if ψ is entailed by the atemporal KB $\langle\mathcal{O},\mathcal{A}_i\rangle$. This can be easily seen by the original semantics of the query (i.e., $\mathcal{O}$ is correctly included/rewritten) and the fact that we have constructed DB such that $\mathsf{DB}\models\mathsf{B}(a,i)$ iff $DB(\mathcal{A}_i)\models B(a)$, $B\in\mathsf{BC}(\mathcal{O})$, and correspondingly for all relations R, $R\in\mathsf{N_R}(\mathcal{O})$. We thus can state the following proposition.

Proposition 6.4. *$\langle\mathcal{O},\mathcal{A}_i\rangle\models\psi$ iff $\mathsf{DB}\models\mathsf{pr}_{(\psi,\mathcal{O})}(i)$.*

We second consider the sets $\mathcal{B}^j_{\mathsf{R|o}}$, $0\le j\le\ell$, where $\ell:=|\mathsf{BC_R}(\mathcal{O})|$, in addition to the input ABoxes $\mathcal{A}_i$, and define the first-order queries $\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{B}^j_{\mathsf{R|o}})}(i)$. Specifically, we set

$$\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{B}^0_{\mathsf{R|o}})}(i):=\mathsf{pr}_{(\psi,\mathcal{O})}(i),$$

and $\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{B}^{j+1}_{\mathsf{R|o}})}(i), 0\le j<\ell$, is obtained from $\mathsf{PerfectRef}(\psi,\mathcal{O})$ by replacing all atoms $B(t_1)$, $B\in\mathsf{BC_R}(\mathcal{O})$, by

$$\mathsf{B}(t_1,i)\vee\bigwedge_{a\in\mathsf{N_I}(\phi)}(t_1\neq a)\wedge\exists p.\mathsf{pr}_{(B(t_1),\mathcal{O}|\mathcal{B}^j_{\mathsf{R|o}})}(p),$$

all flexible basic concept atoms $B(t_1), B\notin\mathsf{BC_R}(\mathcal{O})$, by $\mathsf{B}(t_1,i)$, and all role atoms $R(t_1,t_2), R\in\mathsf{N_R}(\mathcal{O})$, by $\mathsf{R}(t_1,t_2,i)$. The adapted queries decide if a CQ ψ is entailed by the atemporal KBs $\langle\mathcal{O},\mathcal{B}^j_{\mathsf{R|o}}\cup\mathcal{A}_i\rangle$, which is captured by the following proposition.

Proposition 6.5. *For $0\le j<\ell$, $\langle\mathcal{O},\mathcal{B}^j_{\mathsf{R|o}}\cup\mathcal{A}_i\rangle\models\psi$ iff $\mathsf{DB}\models\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{B}^j_{\mathsf{R|o}})}(i)$.*

Proof. The proof is by induction. In particular, the base case where $j=0$ directly follows from the definitions of $\mathcal{B}^0_{\mathsf{R|o}}$ and $\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{B}^0_{\mathsf{R|o}})}$, by Proposition 6.4. In what follows, we assume $j>0$.

Further, note that we construct the rewriting based on $\mathsf{PerfectRef}$ and thus have $\langle \mathcal{O}, \mathcal{B}^j_{\mathsf{R|o}} \cup \mathcal{A}_i \rangle \models \psi$ iff $DB(\mathcal{B}^j_{\mathsf{R|o}} \cup \mathcal{A}_i) \models \mathsf{PerfectRef}(\psi, \mathcal{O})$ [BAC10].

For ($\Rightarrow$) and $DB(\mathcal{B}^j_{\mathsf{R|o}} \cup \mathcal{A}_i) \models \mathsf{PerfectRef}(\psi, \mathcal{O})$, we have a homomorphism π of $\mathsf{PerfectRef}(\psi, \mathcal{O})$ into $DB(\mathcal{B}^j_{\mathsf{R|o}} \cup \mathcal{A}_i)$ and a CQ ψ' in the UCQ $\mathsf{PerfectRef}(\psi, \mathcal{O})$ such that, for all atoms α in ψ', $\pi(\alpha) \in \mathcal{B}^j_{\mathsf{R|o}} \cup \mathcal{A}_i$, by the semantics and the definition of DB. We subsequently show that, for all atoms α in ψ', we have that π is a homomorphism of the corresponding replacement into DB.[9]

If α is a role atom $R(t_1, t_2)$ or flexible basic concept atom $B(t_1)$, then $\pi(\alpha) \notin \mathcal{B}^j_{\mathsf{R|o}}$, by the definition of $\mathcal{B}^j_{\mathsf{R|o}}$; hence, we have $\pi(\alpha) \in \mathcal{A}_i$, by the assumption and the definition of DB. Thus, we directly get that R^{DB} contains the tuple $(\pi(t_1), \pi(t_2), i)$ or that, respectively, B^{DB} contains the tuple $(\pi(t_1), i)$, by the definition of DB. Thus, π is as required.

We consider α to be a rigid basic concept atom $B(t_1)$; hence, it is replaced during the rewriting. If $\pi(\alpha) \in \mathcal{A}_i$, then we get that B^{DB} contains the tuple $(\pi(t_1), i)$, by the definition of DB, such that π is as required.

If $\pi(\alpha) \in \mathcal{B}^j_{\mathsf{R|o}}$, then the definition of $\mathcal{B}^j_{\mathsf{R|o}}$ yields that $\pi(t_1) \notin \mathsf{N_I}(\phi)$ and that there is a $k, 0 \leq k \leq n$, such that $\langle \mathcal{O}, \mathcal{B}^{j-1}_{\mathsf{R|o}} \cup \mathcal{A}_k \rangle \models \pi(\alpha)$. By the induction hypothesis, we thus get $\mathsf{DB} \models \mathsf{pr}_{(\pi(\alpha), \mathcal{O}|\mathcal{B}^{j-1}_{\mathsf{R|o}})}(k)$ from the latter, which together with the former yields that π is as required.

For ($\Leftarrow$), we can argument correspondingly, by considering a given homomorphism π and a satisfied disjunct of $\mathsf{pr}_{(\psi, \mathcal{O}|\mathcal{B}^j_{\mathsf{R|o}})}(i)$, which itself is a conjunction (i.e., a conjunction that originally had been a CQ obtained from $\mathsf{PerfectRef}$, where the atoms were replaced during the rewriting). We consider an arbitrary conjunct therein and show that π is also a homomorphism of that conjunct into $DB(\mathcal{B}^j_{\mathsf{R|o}} \cup \mathcal{A}_i)$. If it (α) replaces a role atom $R(t_1, t_2)$ or flexible basic concept atom $B(t_1)$, then it is of the form $\mathsf{R}(t_1, t_2, i)$ or, respectively, $\mathsf{B}(t_1, i)$. By the assumption that $\mathsf{DB} \models \pi(\alpha)$ and the definition of DB, we then directly get that $R(\pi(t_1), \pi(t_2)) \in \mathcal{A}_i$ or, respectively, $B(\pi(t_1)) \in \mathcal{A}_i$. Hence, π is as required w.r.t. the original atom.

We next consider the disjunction representing a replacement of a rigid basic concept atom $B(t_1)$. If the first disjunct $\mathsf{B}(t_1, i)$ is satisfied, we can argument as in the previous case. Otherwise, we have $\mathsf{DB} \models \exists p.\mathsf{pr}_{(B(\pi(t_1)), \mathcal{O}|\mathcal{B}^{j-1}_{\mathsf{R|o}})}(p)$ and $\pi(t_1) \notin \mathsf{N_I}(\phi)$, which yields $\langle \mathcal{O}, \mathcal{B}^{j-1}_{\mathsf{R|o}} \cup \mathcal{A}_p \rangle \models B(\pi(t_1))$, by the induction hypothesis. Then, we have $B(\pi(t_1)) \in \mathcal{B}^j_{\mathsf{R|o}}$, by the definition of the latter, and again obtain that π is as required. □

[9]We assume the notion of homomorphism to be extended to the replacements, which may be nested disjunctions of rewritings and equality assertions, in the obvious way.

Lastly, we consider the ABox $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ in addition to the input ABoxes and define the queries $\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(i)$. We again start with $\mathsf{PerfectRef}(\psi,\mathcal{O})$ and then replace all flexible atoms $B(t_1)$, $R(t_1,t_2)$ in $\mathsf{PerfectRef}(\psi,\mathcal{O})$ by $\mathsf{B}(t_1,i)$ and $\mathsf{R}(t_1,t_2,i)$, respectively. All rigid atoms $B(t_1)$, $R(t_1,t_2)$ are, respectively, replaced by the following:

$$\left(\bigvee_{B(a)\in\mathcal{B}_{\mathsf{R}|\phi}}(t_1=a)\right)\vee\mathsf{pr}_{(B(t_1),\mathcal{O}|\mathcal{B}_{\mathsf{R}|\mathsf{o}})}(i)\vee\left(\bigvee_{\substack{B=\exists R,a\in\mathsf{N_I}(\mathcal{K}),\\ R(a,e)\in\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}}}(t_1=a)\right)\text{ and}$$

$$\exists p.\mathsf{pr}_{(R(t_1,t_2),\mathcal{O})}(p)\vee\left(\bigvee_{R(a,b)\in\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}}(t_1=a)\wedge(t_2=b)\right).$$

Given the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ it can be readily checked that the adapted query decides if ψ is entailed by the KB $\langle\mathcal{O},\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i\rangle$. This is captured by the following proposition.

Proposition 6.6. *$\langle\mathcal{O},\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i\rangle\models\psi$ iff $\mathsf{DB}\models\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(i)$.*

Proof. First, note that we construct the rewriting based on $\mathsf{PerfectRef}$ and have $\langle\mathcal{O},\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i\rangle\models\psi$ iff $DB(\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i)\models\mathsf{PerfectRef}(\psi,\mathcal{O})$, by the definition of $\mathsf{PerfectRef}$.

For ($\Rightarrow$), we assume $\langle\mathcal{O},\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i\rangle\models\psi$ and hence have a disjunct ψ' in the UCQ $\mathsf{PerfectRef}(\psi,\mathcal{O})$ and a homomorphism π of ψ' into $DB(\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i)$, i.e., for all atoms α in ψ', $\pi(\alpha)\in\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i$. If the atom is flexible, then we get $\pi(\alpha)\in\mathcal{A}_i$. But then, π also fits our rewriting for that case, which addresses the corresponding relation in DB. We consider the case that the atom is rigid. If $\pi(\alpha)\in\mathcal{A}_i$, then by Propositions 6.4 and 6.5 we know that $\mathsf{DB}\models\mathsf{pr}_{(\pi(\alpha),\mathcal{O}|\mathcal{B}_{\mathsf{R}|\mathsf{o}})}(i)$ or $\mathsf{DB}\models\mathsf{pr}_{(\pi(\alpha),\mathcal{O})}(i)$, depending on the kind of atom, and hence our rewriting is as required for this case. If $\pi(\alpha)\in\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, then it can be readily checked that the rewriting covers all parts referenced in the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, again using Propositions 6.4 and 6.5.

For ($\Leftarrow$), we can argument correspondingly, by considering a given homomorphism π and a satisfied disjunct in the UCQ $\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(i)$, which itself is a conjunction (i.e., a CQ obtained from $\mathsf{PerfectRef}$ where the atoms are replaced according to our above construction). We consider an arbitrary such replacement for an atom α, which must be a disjunction, by the definition of the replacements; and at least one disjunct must be satisfied under π. If this disjunct refers to an assertion of $\mathcal{B}_{\mathsf{R}|\phi}$, $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$, or $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}$, then our construction yields that $\pi(\alpha)\in\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, by the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. If the disjunct refers to the rewriting $\mathsf{pr}_{(\alpha,\mathcal{O})}$, then Proposition 6.4 yields that $\pi(\alpha)\in\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. If the disjunct refers to the rewriting $\mathsf{pr}_{(\alpha,\mathcal{O}|\mathcal{B}_{\mathsf{R|o}})}$, then α is a rigid basic concept atom $B(t)$ and Proposition 6.5 yields that $\langle\mathcal{O},\mathcal{B}_{\mathsf{R|o}}\cup\mathcal{A}_i\rangle\models B(\pi(t))$. Since $\mathcal{B}_{\mathsf{R|o}}\subseteq\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, we also have $\langle\mathcal{O},\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i\rangle\models B(\pi(t))$. Then, Lemma 6.2 leads to $B(\pi(t))\in\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. Thus, π is also a homomorphism of $\mathsf{PerfectRef}(\psi,\mathcal{O})$ into $DB(\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_i)$. $\square$

The Final Rewriting

We finally come to the rewriting we target. Our goal is to rewrite a given CQ ψ in a way that the answers to the rewriting of ψ over the ABoxes $\mathcal{A}_i$, $0\leq i\leq n$, captured by DB, actually represent the answers to ψ w.r.t. the KB $\langle\mathcal{O},\mathcal{A}^\dagger_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}\rangle$, with

$$\mathcal{A}^\dagger_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}:=\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}\cup\mathcal{A}_{Q_{\iota(i)}}\cup\mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}\cup\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}\cup\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}\cup\mathcal{A}_i.$$

That is, we want to adopt the approach proposed by Lemma 6.3. In particular, it allows us to again adapt the original UCQs $q_{unsat(\mathcal{O})}$ and $\mathsf{PerfectRef}(\psi,\mathcal{O})$, considering all the above ABoxes. Note that, whereas $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}$, $\mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}$, $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$, and $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$ are of constant size, $\mathcal{A}_{Q_{\iota(i)}}$ and $\mathcal{A}_i$ depend on the considered time point i.

We now start from the queries $q^\dagger_{unsat(\mathcal{O})}$ and $\mathsf{PerfectRef}(\psi,\mathcal{O})^\dagger$, where $\cdot^\dagger$ represents a function that applies the rewriting $\cdot^\dagger$ to every CQ in the given UCQ. We then adapt these queries to finally propose the rewritings $\mathsf{pr}_{unsat(\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$ and $\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$, given arbitrary (constant) sets $\mathcal{S}\subseteq 2^{\{p_1,\dots,p_m\}}$, $X\in\mathcal{S}$, and $\mathcal{B}_\phi\subseteq\{B(a)\mid B\in\mathsf{BC}(\mathcal{O}),\ a\in\mathsf{N_I}(\phi)\}$.[10]

Let thus such sets $\mathcal{S}$, X, and $\mathcal{B}_\phi$ be given. Note that, while the database adressed by the original UCQs $q_{unsat(\mathcal{O})}$ and $\mathsf{PerfectRef}(\psi,\mathcal{O})$ contains only the individuals occurring in the input ABoxes, we need to consider auxiliary elements from $\mathsf{N_I^{aux}}$, $\mathsf{N_I^{tree}}$, and $\mathsf{N_I^{pro}}$. Hence, we especially have to take care of the quantifiers in $q_{unsat(\mathcal{O})}$ and $\mathsf{PerfectRef}(\psi,\mathcal{O})$, which quantify only over the individuals in the original database. To this end, we consider each disjunct

$$q=\exists x_1,\dots,x_\ell.\varphi(x_1,\dots,x_\ell)\wedge\varphi_{\mathsf{filter}}(x_1,\dots,x_\ell)$$

contained in the original queries (i.e., because of the filter condition included by $\cdot^\dagger$, we do not have UCQs anymore), where $\varphi(x_1,\dots,x_\ell)$ is a conjunction of atoms. We then duplicate q several times such that we have 2^ℓ versions $q_0,\dots,q_{2^\ell-1}$ of it, and then augment the quantification of variables to consider also the elements in $\mathsf{N_I^{aux}}\cup\mathsf{N_I^{tree}}\cup\mathsf{N_I^{pro}}$, which however are of constant size, in an appropriate way. Finally, we replace q by the disjunction $q_0\vee\dots\vee q_{2^\ell-1}$.

[10] Note that we here use the subscript part $\mid\mathcal{S},X,\mathcal{B}_\phi$ to describe the dependence on the given parameters, whereas we used the additionally included ABoxes, above. For brevity, we do not use $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}\cup\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}\cup\mathcal{A}_{Q_{\iota(i)}}\cup\mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}\cup\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}\cup\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$, here.

Formally, we consider each quantified variable x_j in every q_k where

$$k = b_0 * 2^0 + \ldots + b_j * 2^j + \ldots + b_{\ell-1} * 2^{\ell-1},$$

and drop the quantification if $b_j = 0$. We then define

$$q_k := \exists' x_1. \ldots \exists' x_\ell . \mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\varphi\wedge\varphi_{\mathsf{filter}}}(i),$$

where $\exists' x_j$ stands for $\exists x_j$ if $b_j = 1$, and for $\bigvee_{a_j \in \mathsf{N}_\mathsf{I}^{\mathsf{aux}} \cup \mathsf{N}_\mathsf{I}^{\mathsf{tree}} \cup \mathsf{N}_\mathsf{I}^{\mathsf{pro}}}$ if $b_j = 0$. For any $b \in \{0,1\}$, we denote by $Q_b(k)$ the set $\{x_j \mid 0 \leq j \leq \ell - 1, b_j = b\}$. The formula $\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\varphi\wedge\varphi_{\mathsf{filter}}}(i)$ is constructed by replacing each atom α in $\varphi \wedge \varphi_{\mathsf{filter}}$ by $\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\alpha}(i)$, which we define depending on the form of α:

- For all $B \in \mathsf{BC}(\mathcal{O})$, $t \in \mathsf{N}_\mathsf{I}(q) \cup \mathsf{N}_\mathsf{V}(q)$:

$$\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|B(t)}(i) :=$$

$$\begin{cases}
 & \text{if } t \in Q_0(k), t = x_j, \\
\mathsf{true} & \quad \text{if } a_j \in \mathsf{N}_\mathsf{I}^{\mathsf{aux}}, \\
 & \quad\quad B(a_j) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X}, \\
\mathsf{true} & \quad \text{if } a_j \in \mathsf{N}_\mathsf{I}^{\mathsf{tree}}, \\
 & \quad\quad B(a_j) \in \mathcal{A}_{R_{\mathsf{F}|\mathsf{aux}[\mathcal{S}]}} \cup \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}, \\
\exists x. \left(\bigwedge_{a \in \mathsf{N}_\mathsf{I}(\phi)} (x \neq a) \right) \wedge \exists p. \mathsf{pr}_{(\exists y.S(x,y), \mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(p) & \quad \text{if } a_j = a_{[S]S\varrho} \in \mathsf{N}_\mathsf{I}^{\mathsf{pro}}, \\
 & \quad\quad \langle \emptyset, \mathcal{A}_{\exists S} \rangle \models B(a_j), \\
\mathsf{false} & \quad \text{otherwise}; \\
 & \text{if } t \in Q_1(k) \cup \mathsf{N}_\mathsf{I}(q), \\
\left(\bigvee_{\substack{B(a) \in \mathcal{A}_{Q_X}, \\ a \in \mathsf{N}_\mathsf{I}(\phi)}} (t = a) \right) \vee \mathsf{B}(t,i) & \quad \text{if } B \notin \mathsf{BC}_\mathsf{R}(\mathcal{O}), \\
\left(\bigvee_{\substack{B(a) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}, \\ a \in \mathsf{N}_\mathsf{I}(\phi)}} (t = a) \right) \vee \mathsf{pr}_{(B(t), \mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(i) & \quad \text{otherwise};
\end{cases}$$

- For all $R \in \mathsf{N}_\mathsf{R}^-(\mathcal{O})$, $t_1, t_2 \in \mathsf{N}_\mathsf{I}(q) \cup \mathsf{N}_\mathsf{V}(q)$:

$$\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|R(t_1,t_2)}(i) :=$$

$$\begin{cases}
 & \text{if } t_1, t_2 \in Q_0(k), t_1 = x_{j_1}, t_2 = x_{j_2}, \\
 & \quad \text{if } a_{j_1}, a_{j_2} \notin \mathsf{N}_\mathsf{I}^{\mathsf{pro}}, \\
\mathsf{true} & \qquad \text{if } R(a_{j_1}, a_{j_2}) \in \\
 & \qquad\quad \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X} \cup \mathcal{A}_{R_{\mathsf{F}|\mathsf{aux}[\mathcal{S}]}}, \\
\mathsf{true} & \qquad \text{if } R(a_{j_1}, a_{j_2}) \in \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}, \\
\mathsf{false} & \qquad \text{otherwise;} \\
\exists x. \left(\bigwedge_{a \in \mathsf{N}_\mathsf{I}(\phi)} (x \neq a) \right) \wedge \exists p. \mathsf{pr}_{(\exists y.S(x,y), \mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(p) & \quad \text{if } a_{j_1} = a_{[S]\varrho} \in \mathsf{N}_\mathsf{I}^{\mathsf{pro}}, \\
 & \qquad R(a_{j_1}, a_{j_2}) \in \mathcal{A}_{\exists S}, \\
\mathsf{false} & \quad \text{otherwise;} \\
 & \text{if } t_1 \in Q_0(k), t_2 \in \mathsf{N}_\mathsf{I}(q), t_1 = x_j, \\
\mathsf{true} & \quad \text{if } R(a_j, t_2) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X}, \\
\mathsf{true} & \quad \text{if } R(a_j, t_2) \in \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}, \\
\mathsf{false} & \quad \text{otherwise;} \\
 & \text{if } t_1 \in Q_0(k), t_2 \in Q_1(k), t_1 = x_j, \\
\bigvee_{\substack{R(a_j,b) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X}, \\ b \in \mathsf{N}_\mathsf{I}(\phi)}} (t_2 = b) & \quad \text{if } a_j \in \mathsf{N}_\mathsf{I}^{\mathsf{aux}}, \\
\bigvee_{\substack{R(a_j,b) \in \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}, \\ b \in \mathsf{N}_\mathsf{I}(\phi)}} (t_2 = b) & \quad \text{if } a_j \in \mathsf{N}_\mathsf{I}^{\mathsf{tree}}, \\
\bigwedge_{b \in \mathsf{N}_\mathsf{I}(\phi)} (t_2 \neq b) \wedge \exists p. \mathsf{pr}_{(\exists y.S(t_2,y), \mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(p) & \quad \text{if } a_j = a_{[S]S} \in \mathsf{N}_\mathsf{I}^{\mathsf{pro}}, \\
 & \qquad R(a_j, [S]) \in \mathcal{A}_{\exists S}, \\
\mathsf{false} & \quad \text{otherwise;} \\
 & \text{if } t_1, t_2 \in Q_1(k) \cup \mathsf{N}_\mathsf{I}(q), \\
\left(\bigvee_{\substack{R(a,b) \in \mathcal{A}_{Q_X}, \\ a,b \in \mathsf{N}_\mathsf{I}(\phi)}} (t_1 = a) \wedge (t_2 = b) \right) \vee \mathsf{R}(t_1, t_2, i) & \quad \text{if } R \notin \mathsf{N}_{\mathsf{RR}}, \\
\left(\bigvee_{\substack{R(a,b) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}, \\ a,b \in \mathsf{N}_\mathsf{I}(\phi)}} (t_1 = a) \wedge (t_2 = b) \right) \vee \mathsf{pr}_{(R(t_1,t_2), \mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(i) & \quad \text{otherwise;}
\end{cases}$$

- For all $t \in \mathsf{N}_\mathsf{I}(q) \cup \mathsf{N}_\mathsf{V}(q)$:

$$\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\mathsf{pro}(t)}(i) := \begin{cases} \mathsf{true} & \text{if } t \in Q_0(k), t = x_j, a_j \in \mathsf{N}_\mathsf{I}^{\mathsf{pro}}, \\ \mathsf{false} & \text{otherwise;} \end{cases}$$

- For all $t_1, t_2 \in \mathsf{N_I}(q) \cup \mathsf{N_V}(q)$:

$$\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|t_1=t_2}(i) := \begin{cases} t_1 = t_2 & \text{if } t_1, t_2 \in Q_1(k) \cup \mathsf{N_I}(q), \\ \mathsf{true} & \text{if } t_1, t_2 \in Q_0(k),\, t_1 = x_{j_1},\, t_2 = x_{j_2},\, a_{j_1} = a_{j_2}, \\ \mathsf{false} & \text{otherwise;} \end{cases}$$

Note that all parts above referencing $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}, \mathcal{A}_{Q_X}, \mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}$, and $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$ do not depend on the input ABoxes in any way. The 'replacement' definitions basically consider two different cases, depending on whether the atom contains a variable which is now unquantified. If this is the case, corresponding assertions can only occur within $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X} \cup \mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}} \cup \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}} \cup \mathcal{A}_{R_{\mathsf{F|o}}}$; otherwise, we also take the input ABoxes and $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ into account—by using the auxiliary query $\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}$. Note that variables that are now unquantified but associated to an element from $\mathsf{N_I^{pro}}$ present an exception because we also consider the input ABoxes then. This is because we need to ensure that prototypical elements as considered actually occur in $\mathcal{A}^\dagger_{R_{\mathsf{F|o}}}$.

Further, recall that for $i = -1$, the same definitions apply, but the underlying ABox $\mathcal{A}_i$ is then empty.

The next lemma establishes the correctness of our translation of the original UCQs over $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, into FO-formulas over DB.

Lemma 6.7. *For every CQ ψ occurring in ϕ, and all $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$, $X \in \mathcal{S}$, $\mathcal{B}_\phi \subseteq \bigcup_{B \in \mathsf{BC}(\mathcal{O}), a \in \mathsf{N_I}(\phi)} \{B(a)\}$, and $i \in \{-1, 0, \dots, n\}$, we have:*

- $DB(\mathcal{A}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}) \models q_{unsat(\mathcal{O})}$ *iff* $\mathsf{DB} \models \mathsf{pr}_{unsat(\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$.
- $DB(\mathcal{A}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}) \models \mathsf{PerfectRef}(\psi, \mathcal{O})$ *iff* $\mathsf{DB} \models \mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$,

Proof. For ($\Rightarrow$), we assume $DB(\mathcal{A}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}) \models q'$, where q' is one of the two UCQs. By Lemma 6.3 and the semantics of disjunction, we can equivalently consider the ABox $\mathcal{A}^\dagger_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}$ and assume $DB(\mathcal{A}^\dagger_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}) \models (q')^\dagger$. We thus have a homomorphism π of some query

$$q^\dagger = \exists x_1, \dots, x_\ell . \varphi(x_1, \dots, x_\ell) \wedge \varphi_{\mathsf{filter}}(x_1, \dots, x_\ell)$$

contained in $(q')^\dagger$ into $DB(\mathcal{A}^\dagger_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}})$, where $\varphi(x_1, \dots, x_\ell)$ is a conjunction of atoms. Further, we must have a query q_k that is an adaptation of $q^\dagger$ in our translation of $q^\dagger$ with

$$k = b_0 * 2^0 + \ldots + b_j * 2^j + \ldots + b_{\ell-1} * 2^{\ell-1},$$

where $b_j = 0$ iff $\pi(x_j) \in \mathsf{N_I^{aux}} \cup \mathsf{N_I^{tree}} \cup \mathsf{N_I^{pro}}$, $0 \leq j \leq \ell - 1$.

We now show that π is also a homomorphism of q_k into DB if restricted to $\mathsf{N_V}(q_k) \cup \mathsf{N_I}(q_k)$.[11] For that, we first consider all conjuncts α of q_k, and thus the replacements $\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\alpha}(i)$ that were introduced by our adaptation. By the assumption, we have that $DB(\mathcal{A}^{\dagger}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}) \models \pi(\alpha)$, where we $\pi(\alpha)$ denotes the assertion obtained by replacing the variable(s) x in the atom α by $\pi(x)$. We thus get that $\pi(\alpha) \in \mathcal{A}^{\dagger}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}}$, by the definition of $DB(\mathcal{A}^{\dagger}_{\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}})$. As mentioned above, the definitions of the replacement formulas basically depend on whether the regarded atom contains a variable x_j, $0 \leq j \leq \ell - 1$, such that $\pi(x_j) \in \mathsf{N_I^{aux}} \cup \mathsf{N_I^{tree}} \cup \mathsf{N_I^{pro}}$. Let α now be an arbitrary atom containing a variable x_j. First note that, if $\alpha = \mathsf{pro}(x_j)$, the replacement is in accordance with the semantics of the predicate pro (i.e., especially, pro must evaluate to false for any $\pi(y)$ with y being a quantified variable) such that π fits in this regard. Moreover, note that, after our translation, $\varphi_{\mathsf{filter}}$ is a conjunction of equality statements. We now consider the other kinds of atoms α.

- We first assume $\pi(x_j) \in \mathsf{N_I^{aux}} \cup \mathsf{N_I}(\mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}})$. That is, we also regard elements from $\mathsf{N_I^{tree}}$ of the form $a_{a_x\varrho}$, where $a_x \in \mathsf{N_I^{aux}}$. We hence must have $\pi(\alpha) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X} \cup \mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}$, because the input ABoxes and $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ only regard elements of $\mathsf{N_I}(\mathcal{K})$, and neither $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$ nor $\mathcal{A}^{\dagger}_{R_{\mathsf{F|o}}}$ contain elements of this kind. For the case that $\pi(x_j) \in \mathsf{N_I^{tree}}$, we obtain $\pi(\alpha) \in \mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}$, by similar arguments.

  By the definition of $\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\alpha}(i)$, we directly get that $\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\alpha}(i) = \mathsf{true}$ if α does not contain a variable that is quantified within q_k. To see this, observe that π does not map to elements of $\mathsf{N_I^{pro}}$, $\mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}}$ does not contain basic concept assertions on elements of $\mathsf{N_I}(\mathcal{K})$, nor does it contain role assertions with named individuals (i.e., referring to those in $\mathsf{N_I}(q_k)$).

  The latter argument is also important for the case that α contains a variable y that is quantified within q_k; and hence we have $\pi(y) \in \mathsf{N_I}(\mathcal{K})$. We thus obtain $\pi(\alpha) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X}$, for this case. In particular, α must be of the form $R(x_j, y), R \in \mathsf{N_R^-}$. But $\pi(\alpha) = R(\pi(x_j), \pi(y))$ and $R(\pi(x_j), \pi(y)) \in \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X}$ then yield that $\mathsf{rep}_{\mathcal{S},X,\mathcal{B}_\phi|\alpha}(i)$ contains the disjunct $(y = \pi(y))$, for which π is obviously as required.

- We now assume $\pi(x_j) \in \mathsf{N_I^{tree}} \cap \mathsf{N_I}(\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}})$. That is, we regard elements from $\mathsf{N_I^{tree}}$ of the form $a_{b\varrho}$, where $b \in \mathsf{N_I}(\phi)$—the elements from $\mathsf{N_I^{tree}}$ that remain to be considered. We hence must have $\pi(\alpha) \in \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$, by arguments similar to those in the previous item, and also the part remaining can be shown correspondingly.

- If $\pi(x_j) \in \mathsf{N_I^{pro}}$, then we obviously have $\pi(\alpha) \in \mathcal{A}^{\dagger}_{R_{\mathsf{F|o}}}$. Let a_j be of the form

[11] Note that we did not define the notion of homomorphism w.r.t. disjunction ($\vee$) and (in)equality predicates in Section 2.2. But the corresponding extension should be obvious.

$a_{[S]\varrho}$. Then, $\pi(\alpha)$ must come from $\mathcal{A}_{\exists S}$, and the latter must be part of $\mathcal{A}^{\dagger}_{R_{\mathsf{F|o}}}$, by the definition of $\cdot^{\dagger}$. The latter also yields that we have an individual $a \in \mathsf{N_I}(\mathcal{K}) \setminus \mathsf{N_I}(\phi)$ such that $\mathcal{A}_{\exists S(a)} \subseteq \mathcal{A}_{R_{\mathsf{F|o}}}$. We thus have some p such that $\langle \mathcal{O}, \mathcal{B}_{\mathsf{R|o}} \cup \mathcal{A}_p \rangle \models \exists S(a)$, by the definition of $R_{\mathsf{F|o}}$, which, in turn, yields that the replacement is satisfied by the definition of $\mathsf{pr}_{(\exists y.S(x,y),\mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(p)$, which contains the disjunct $\mathsf{pr}_{(\exists y.S(x,y),\mathcal{O}|\mathcal{B}_{\mathsf{R|o}})}(p)$, and Proposition 6.5.

It thus remains to consider the case where α contains a variable y that is quantified within q_k and hence $\pi(y) \in \mathsf{N_I}(\mathcal{K})$. Then, α must be of the form $R(x_j, y), R \in \mathsf{N_R^-}$, and $\mathcal{A}^{\dagger}_{R_{\mathsf{F|o}}}$ must contain the assertion $R(a_j, \pi(y))$. But we then have also $R(a_j, [S]) \in \mathcal{A}_{\exists S}$ (and $\varrho = S$), $S \in \mathsf{N_R^-}$. Moreover, it can be readily checked that the formula which is then proposed as replacement evaluates to true because we have that $\pi(y) \notin \mathsf{N_I}(\phi)$, using the same arguments as in the previous case.

- We consider the case where the variable(s) in α are not mapped to our auxiliary elements of $\mathsf{N_I^{aux}} \cup \mathsf{N_I^{tree}} \cup \mathsf{N_I^{pro}}$ (i.e., they are mapped to elements of $\mathsf{N_I}(\mathcal{K})$) and first assume α to be a flexible atom. We then must have $\pi(\alpha) \in \mathcal{A}_{Q_X} \cup \mathcal{A}_i$. If $\pi(\alpha) \in \mathcal{A}_{Q_X}$, we can argue similarly to the first case. The other case, $\pi(\alpha) \in \mathcal{A}_i$, is also covered by the definition. Here, the definition of DB based on $\mathcal{A}_i$ yields that π is as required (e.g., for $\alpha = R(x_l, a), a \in \mathsf{N_I}(\mathcal{K})$, we have $\mathsf{DB} \models \mathsf{R}(\pi(x_l), a, i)$ iff $R(\pi(x_l), a) \in \mathcal{A}_i$).

  If α is rigid, then we have $\pi(\alpha) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_{Q_{\mathsf{R}}} \cup \mathcal{A}_{Q_X} \cup \mathcal{A}_i$. Since the atom is rigid, we have that if $\pi(\alpha) \in \mathcal{A}_{Q_X}$, then $\pi(\alpha) \in \mathcal{A}_{Q_{\mathsf{R}}}$; and the latter ABox is addressed directly in the replacement. The other cases are covered by $\mathsf{pr}_{(\alpha,\mathcal{O}|\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]})}(i)$, according to Proposition 6.6.

Given that π thus is a homomorphism of all our replacements into DB, we obtain $\mathsf{DB} \models q_k$, which leads to $\mathsf{DB} \models q''$, our translation of the queries for which this was to be shown.

We only sketch the proof for the direction ($\Leftarrow$) since it works similarly; it also does not differ for the two items. By the semantics and the definition of the rewritings, we have an adaptation q_k in the given rewriting q'' that is an adaptation of a query $q^{\dagger}$ in $(q')^{\dagger}$ and satisfied in DB. We thus have a homomorphism π of q_k into DB (i.e., w.r.t. the individual names and (existentially quantified) variables in q_k) and show that we can extend it adequately to cover the terms occurring in $q^{\dagger}$. By our construction, q_k is a disjunction; hence, by the semantics, we have that one of these disjuncts is satisfied. We regard the individual names a_j from $\mathsf{N_I^{aux}} \cup \mathsf{N_I^{tree}} \cup \mathsf{N_I^{pro}}$ associated to the variables x_j in $q^{\dagger}$ and to that disjunct. In particular, we extend π such that $\pi(x_j) = a_j$ for all these variables, and subsequently show that this definition satisfies our purpose. We now consider an arbitrary conjunct representing the replacement for an atom $B(t)$ in $q^{\dagger}$ in the disjunct under consideration (ignoring the conjunct of the form $\varphi_{\mathsf{filter}}$ for now).

If $\pi(t) \in \mathsf{N}_\mathsf{I}^{\mathsf{aux}} \cup \mathsf{N}_\mathsf{I}^{\mathsf{tree}}$, then it can be readily checked that π is as required w.r.t. $B(t)$, by the definition of DB and the semantics. If $\pi(t) \in \mathsf{N}_\mathsf{I}^{\mathsf{pro}}$, we additionally need to take Proposition 6.6 and Lemma 6.2 into account. The definition of DB together with Proposition 6.6 lastly confirm the case where $\pi(t) \in \mathsf{N}_\mathsf{I}(\mathcal{K})$. The proof is correspondingly for role atoms.

We lastly consider $\varphi_{\mathsf{filter}}$, which itself is a conjunction of implications. By our extension of π in accordance with the individual names associated to the disjunct under consideration, the definition of the predicate pro, and that of the replacement of the corresponding atoms, we have that each atom occurring in $\varphi_{\mathsf{filter}}$ is satisfied under the extended π iff it evaluates to true in the rewritten form in DB. Hence, π is as required. □

Rewriting r-satisfiability

Based on the previous observations, we now can define the following FO-formulas, for all $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$, $X \in \mathcal{S}$, and $\mathcal{B}_\phi \subseteq \{B(a) \mid B \in \mathsf{BC}(\mathcal{O}),\ a \in \mathsf{N}_\mathsf{I}(\phi)\}$ (cf. Definition 4.8):

- $f_{(\text{R1})}(i) := \neg\mathsf{pr}_{unsat(\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$;
- $f_{(\text{R2})}(i) := \bigwedge_{p_\ell \in \overline{X}} \neg\mathsf{pr}_{(\alpha_\ell,\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$;
- $f_{(\text{R5})}(i) := \bigwedge_{\substack{\alpha \in Q_\mathsf{R}^-,\\ \psi \text{ witness query}\\ \text{for } \alpha \text{ w.r.t. } \mathcal{O}}} \neg\mathsf{pr}_{(\psi,\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$;

Lastly, we define the abbreviation

$$\mathsf{rsat}_{\mathcal{S},X}(i) := f_{(\text{R1})}(i) \wedge f_{(\text{R2})}(i) \wedge f_{(\text{R5})}(i),$$

which represents our (yet partial) rewriting of r-satisfiability. We next provide a lemma capturing this intention.

Lemma 6.8. *Let $\mathcal{S} = \{X_1, \dots, X_k\} \subseteq 2^{\{p_1,\dots,p_m\}}$, $\iota\colon \{0,\dots,n\} \to \{1,\dots,k\}$, and $\mathcal{B}_\phi \subseteq \{B(a) \mid B \in \mathsf{BC}(\mathcal{O}),\ a \in \mathsf{N}_\mathsf{I}(\phi)\}$ be arbitrary. The tuple*

$$(\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}, Q_{\mathsf{R}[\mathcal{S}]}, Q^-_{\mathsf{R}[\mathcal{S}]}, R_{\mathsf{F}[\mathcal{S},\mathcal{B}_\phi]}),$$

is r-complete w.r.t. $\mathcal{S}$ and ι iff

- *for all i, $0 \le i \le n$, we have $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},X_{\iota(i)},\mathcal{B}_\phi}(i)$;*
- *for all $X \in \mathcal{S}$, we have $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},X}(-1)$; and*
- *for all $S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O}) \setminus \mathsf{N}_\mathsf{RR}^-$ and $a \in \mathsf{N}_\mathsf{I}(\phi)$, we have $\exists S(a) \in \mathcal{B}_\phi$ iff there is an $i, 0 \le i \le n$, such that $\mathsf{DB} \models \mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},X_{\iota(i)},\mathcal{B}_\phi)}(i)$.*

Proof. We consider Definition 4.8. Given its definition, it can be easily seen that $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ is an ABox type; and by the definitions of $Q_{\mathsf{R}[\mathcal{S}]}$ and $Q^{\neg}_{\mathsf{R}[\mathcal{S}]}$, we obviously have that (R3) and (R4) are satisfied. Hence, it remains to consider Conditions (R1), (R2), (R5), and (R6). Considering Lemma 6.7, it is however easy to see that the formulas in $\mathsf{rsat}_{\mathcal{S},X}(i)$ cover the first three of them adequately. We regard (R6) and first observe that, w.r.t. the ABoxes of (R6), the individual names considered by $R_{\mathsf{F|o}}$ can only occur within $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \bigcup_{0\leq i\leq n} \mathcal{A}_i$, and those of $R_{\mathsf{F|aux}[\mathcal{S}]}$ in $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \bigcup_{X\in\mathcal{S}} \mathcal{A}_{Q_X}$.

- If we have $\exists S(b) \in R_{\mathsf{F|o}}$, then there is an index i, $0 \leq i \leq n$, such that $\langle \mathcal{O}, \mathcal{B}_{\mathsf{R|o}} \cup \mathcal{A}_i\rangle \models \exists S(b)$. By the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, we get $\mathcal{B}_{\mathsf{R|o}} \subseteq \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, and hence also $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i\rangle \models \exists S(b)$.

  Conversely, if $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i\rangle \models \exists S(b)$, then, by Proposition 3.3, this assertion follows from rigid assertions in $\mathcal{B}_{\mathsf{R|o}}$ and several rigid role assertions of the form $R(b,a)$, $R \in \mathsf{N}^-_{\mathsf{R}}(\mathcal{O})$, each of which follows from a single $\mathcal{A}_j$ (and $\mathcal{O}$). But then the corresponding rigid basic concept assertions $\exists R(b)$ must also be in $\mathcal{B}_{\mathsf{R|o}}$, by our construction, and hence $\exists S(b)$ follows exclusively from $\mathcal{B}_{\mathsf{R|o}}$ (and $\mathcal{O}$). But then it is contained in $R_{\mathsf{F|o}}$, by the definition of this set.

- For every $\exists S(a_y) \in R_{\mathsf{F|aux}[\mathcal{S}]}$ for some $a_y \in \mathsf{N}^{\mathsf{aux}}_{\mathsf{I}}$, we directly get from the definition of $R_{\mathsf{F|aux}[\mathcal{S}]}$ that there must be an $X \in \mathcal{S}$ with $\langle \mathcal{O}, \mathcal{A}_{Q_X}\rangle \models \exists S(a_y)$.

  On the other hand, if $\langle \mathcal{O}, \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_X}\rangle \models \exists S(a_y)$ for some $X \in \mathcal{S}$, then by Proposition 3.3 it can only follow from assertions involving a_y. But for a_y there is a unique query $\alpha \in \mathcal{Q}_\phi$ that contains y, and the fact that $a_y \in \mathsf{N}^{\mathsf{aux}}_{\mathsf{I}}$ implies that $\alpha_j \in Q_{\mathsf{R}[\mathcal{S}]}$. Hence, there must be some $X' \in \mathcal{S}$ with $p_j \in X'$, and in particular $\mathcal{A}_{Q_{X'}}$ implies all assertions about a_y in $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}$. This shows that $\exists S(a_y)$ already follows from $\mathcal{A}_{Q_{X'}}$ (and $\mathcal{O}$), and hence $\exists S(a_y) \in R_{\mathsf{F|aux}[\mathcal{S}]}$.

- For $R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$, the definition together with the condition in the last item and Lemma 6.7 yields $\exists S(a) \in R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$ iff there is an $i, 0 \leq i \leq n$, such that $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_i \cup \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}\rangle \models \exists S(a)$ since $R_{\mathsf{F|aux}[\mathcal{S}]}$ and $R_{\mathsf{F|o}}$ cannot be involved here. However, note that all parts of $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$ which are relevant to obtain such a conclusion (cf. Proposition 3.3) are contained in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, by the definition of the latter and that of $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}}$. This means we get $\exists S(a) \in R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$ iff there is an $i, 0 \leq i \leq n$, such that $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_i\rangle \models \exists S(a)$.

On the other hand, it is easy to see that (R6) together with the definition of $R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$ implies that the last condition of the lemma is satisfied. Hence, this condition is equivalent to (R6), which concludes the proof. ☐

However, there is still one piece missing between Lemmas 6.8 and 4.9.

Lemma 6.9. *Let $\mathcal{S}$ and ι be as above. Then there is an r-complete tuple w.r.t. $\mathcal{S}$ and ι iff there is a set $\mathcal{B}_\phi$ such that the tuple*

$$(\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}, Q_{\mathsf{R}[\mathcal{S}]}, Q^{\neg}_{\mathsf{R}[\mathcal{S}]}, R_{\mathsf{F}[\mathcal{S},\mathcal{B}_\phi]})$$

is r-complete w.r.t. $\mathcal{S}$ and ι.

Proof. The direction ($\Leftarrow$) is trivial.

We consider ($\Rightarrow$). Let $(\mathcal{A}_\mathsf{R}, Q_\mathsf{R}, Q^{\neg}_\mathsf{R}, R_\mathsf{F})$, be an r-complete tuple. We need to show that there is a set $\mathcal{B}_\phi$ such that $(\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}, Q_{\mathsf{R}[\mathcal{S}]}, Q^{\neg}_{\mathsf{R}[\mathcal{S}]}, R_{\mathsf{F}[\mathcal{S},\mathcal{B}_\phi]})$ is r-complete as well. We define

$$\mathcal{B}_\phi := \{B(a) \in \mathcal{A}_\mathsf{R} \cup R_\mathsf{F} \mid B \in \mathsf{BC}(\mathcal{O}), a \in \mathsf{N_I}(\phi)\}.$$

We subsequently show that this definition is as required. Our tuple obviously satisfies Conditions (R3) and (R4), by construction. Further observe that, since we constructed $Q_{\mathsf{R}[\mathcal{S}]}$ and $Q^{\neg}_{\mathsf{R}[\mathcal{S}]}$ minimal w.r.t. Conditions (R3) and (R4), and the given tuple also must respect them, we know that $Q_{\mathsf{R}[\mathcal{S}]} \subseteq Q_\mathsf{R}$ and $Q^{\neg}_{\mathsf{R}[\mathcal{S}]} \subseteq Q^{\neg}_\mathsf{R}$, and thus have $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \subseteq \mathcal{A}_{Q_\mathsf{R}}$.

For Condition (R6) and the sets $R_{\mathsf{F|aux}[\mathcal{S}]}$ and $R_{\mathsf{F|o}}$, we can apply the same arguments as in the proof of Lemma 6.8. Since the given tuple satisfies (R6), we have $\exists S(a) \in R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$ iff there is i, $0 \leq i \leq n$, with $\langle \mathcal{O}, \mathcal{A}_\mathsf{R} \cup \mathcal{A}_{Q_\mathsf{R}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_i \rangle \models \exists S(a)$, by our construction of $R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$. Since $\mathcal{B}_\phi$ contains all the rigid basic concepts on elements of $\mathsf{N_I}(\phi)$ which are contained in $\mathcal{A}_\mathsf{R}$, $\mathcal{A}_\mathsf{R}$ is an ABox type, $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \subseteq \mathcal{A}_{Q_\mathsf{R}}$, and the given tuple satisfies (R1) (i.e., the consistency established by (R1) ensures that no other relevant basic concept assertions can be derived from assertions in $\mathcal{A}_\mathsf{R}$ or $\mathcal{A}_{Q_\mathsf{R}}$), Proposition 3.3 yields that $\exists S(a) \in R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}$ iff there is an $i, 0 \leq i \leq n$, such that $\langle \mathcal{O}, \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{Q_{\iota(i)}} \cup \mathcal{A}_i \rangle \models \exists S(a)$. Hence Condition (R6) is also satisfied.

It thus remains to consider Conditions (R1), (R2), and (R5).

Let $\mathcal{K}^i_\mathsf{R}$ be the consistent KB that exists for the given tuple by (R1), and start with Condition (R1). By construction and the arguments above, we know that $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \subseteq \mathcal{A}_{Q_\mathsf{R}}$ and $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}} \subseteq \mathcal{A}_{R_\mathsf{F}}$. For every $\exists S(b) \in R_{\mathsf{F|aux}[\mathcal{S}]}$, we know that it follows already from some $\mathcal{A}_{Q_X}$ with $X \in \mathcal{S}$ (and $\mathcal{O}$), and hence by (R6) it must be contained in R_F. Similarly, if $\exists S(b) \in R_{\mathsf{F|o}}$, then it must follow from $\mathcal{A}_\mathsf{R}$ and some $\mathcal{A}_i$ (since $\mathcal{A}_\mathsf{R}$ must contain all assertions in $\mathcal{B}_{\mathsf{R|o}}$), and hence is contained in R_F. This means that $\mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}} \cup \mathcal{A}_{R_{\mathsf{F|o}}} \cup \mathcal{A}_{R_{\mathsf{F|aux}[\mathcal{S}]}} \subseteq \mathcal{A}_{R_\mathsf{F}}$.

By the construction of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, we further have that all positive assertions contained in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ have to be positive in $\mathcal{A}_\mathsf{R}$, because these assertions are implied by some KB $\langle \mathcal{O}, \mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}} \cup \mathcal{A}_{R_{\mathsf{F}|\phi[\mathcal{B}_\phi]}} \cup \mathcal{A}_i \rangle$ and $\mathcal{K}^i_\mathsf{R}$ is consistent, by assumption. Hence, the only difference left between $\mathcal{K}^i_\mathsf{R}$ and $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ (i.e., focusing on the assertions in $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$; apart from additional assertions in $\mathcal{K}^i_\mathsf{R}$) might be based on some negative rigid assertion(s) in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, which then must occur positively in $\mathcal{A}_\mathsf{R}$ (i.e., because $\mathcal{A}_\mathsf{R}$ is an ABox type), and which leads to the inconsistency of $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$.

To show that this cannot be the case, we provide a model for $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. Since the given tuple satisfies (R1) and $\mathcal{K}^i_\mathsf{R}$ must contain all positive assertions of $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, we get that the KB $[\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}]^+$, obtained by dropping the negative assertions in $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, must be consistent, too. We now assume $\mathcal{I}$ to be the canonical interpretation of that KB.

We start considering a negative role assertion $\neg R(a,b)$ in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. To show that $\mathcal{I} \not\models R(a,b)$, we need to show that none of the ABoxes in $[\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}]^+$ contains a role assertion $S(a,b)$ with $\mathcal{O} \models S \sqsubseteq R$. Consider first the case that S is rigid. Since all rigid assertions that occur in $\mathcal{A}_{Q_{\iota(i)}}$ are also contained in $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}$ $(*)$, the definitions of the ABoxes in $\mathcal{A}_{R_{\mathsf{F}[\mathcal{S},\mathcal{B}_\phi]}}$ (which do not contain assertions with two elements of $\mathsf{N_I}(\mathcal{K})$) and the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ yield that all rigid role assertions that only contain elements of $\mathsf{N_I}(\mathcal{K})$ and occur in some of our ABoxes, $\mathcal{A}_{Q_{\iota(i)}}$ or $\mathcal{A}_{Q_{\mathsf{R}[\mathcal{S}]}}$, are positively contained in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, too. Because $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ is an ABox type (i.e., only one of $S(a,b)$ or $\neg S(a,b)$ is contained in it), this means that $S(a,b)$ cannot occur in one of the ABoxes. Assume now that S is flexible and $S(a,b)$ occurs in $\mathcal{A}_i$ or $\mathcal{A}_{Q_{\iota(i)}}$. By the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ and Definition 4.4, we then must have $R(a,b) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ or $R(a,b) \in Q_{\mathsf{R}[\mathcal{S}]}$. But the latter also implies that $R(a,b) \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, which contradicts our assumption.

Next, we assume $\neg B(a)$ to be a negative rigid basic concept assertion in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. Observe that we have $\neg B(a) \in \mathcal{A}_\mathsf{R}$ if $a \in \mathsf{N_I}(\phi)$, by the definitions of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ and $\mathcal{B}_\phi$. Since $\mathcal{A}_\mathsf{R}$ is an ABox type and the given tuple satisfies (R1), we obtain $\mathcal{I}' \not\models B(a)$, where $\mathcal{I}'$ is the canonical model of $\mathcal{K}^i_\mathsf{R}$, by Proposition 3.2. By our above observation about the positive assertions in $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, this interpretation must also satisfy $[\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}]^+$. But we then must have $\mathcal{I} \not\models B(a)$, by Proposition 3.5.

We consider the case $a \notin \mathsf{N_I}(\phi)$. If $\mathcal{I} \models B(a)$, then there must be some positive concept and role assertions about a in the ABoxes

$$\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}} \cup \mathcal{A}_i, 0 \leq i \leq n,$$

that together imply B, by Proposition 3.3. However, by Lemma 6.2 and the definitions of $R_{\mathsf{F}|\mathsf{o}}$ and $\mathcal{A}_{R_{\mathsf{F}|\mathsf{o}}}$ we have that all relevant assertions within the latter are contained in $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. But then, we only have to consider the ABox $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]} \cup \mathcal{A}_i$, for which we can again apply Lemma 6.2 to obtain $\alpha \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. This contradicts our assumption that $\neg\alpha \in \mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$, by the definition of $\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$. We thus can conclude that the tuple $(\mathcal{A}_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}, Q_{\mathsf{R}[\mathcal{S}]}, Q^{\neg}_{\mathsf{R}[\mathcal{S}]}, R_{\mathsf{F}[\mathcal{S},\mathcal{B}_\phi]})$ satisfies Condition (R1).

We consider the two conditions left. If one of them is contradicted, there must be a homomorphism of one of the considered CQs into the canonical model of $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$.

However, we already mentioned above that all positive assertions contained in one of the ABoxes of $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ must also be contained in the ABox of $\mathcal{K}^i_{\mathsf{R}}$. By the semantics, we thus obtain that any such homomorphism into the canonical model of $\mathcal{K}^i_{\mathsf{R}[\mathcal{S},\mathcal{B}_\phi]}$ would also be a homomorphism into the canonical model of $\mathcal{K}^i_{\mathsf{R}}$. This contradicts the assumption that $\mathcal{K}^i_{\mathsf{R}}$ satisfies Conditions (R1), (R2) and (R5), by Proposition 3.5. □

6.2 An Alternating Logarithmic Time Turing Machine

In this section, we finally provide an alternating Turing Machine (ATM) that solves our problem in logarithmic time. However, before generally describing this type of TM and our specific machine, we first introduce some notation and establish auxiliary results that later facilitate our construction. In what follows, we use the following notation:

- We assume ϕ^{p} to be separated according to Proposition 2.8 and denote by ϕ^{b} its *propositional Boolean abstraction*, which is obtained from ϕ^{p} by replacing all top-level future and past subformulas by propositional variables.
- We assume ϕ^{b} to contain the propositional variables $q_1, \dots, q_o$ in place of the top-level temporal subformulas $f_1, \dots, f_o$ occurring in ϕ^{p}—such that f_i was replaced by q_i, for $1 \leq i \leq o$.
- $\mathcal{P}$ and $\mathcal{F}$ denote the sets of replaced top-level past and future subformulas in ϕ^{p}, respectively, i.e., they form a partition of $\{f_1, \dots, f_o\}$.
- $\mathcal{V}$ denotes the set of all valuations $v\colon \{q_1, \dots, q_o\} \to \{\mathsf{true}, \mathsf{false}\}$ under which ϕ^{b} evaluates to true.
- For each $v \in \mathcal{V}$, we set

  $$\mathcal{P}^v := \{f_i \in \mathcal{P} \mid v(q_i) = \mathsf{true}\} \cup \{\neg f_i \mid f_i \in \mathcal{P},\ v(q_i) = \mathsf{false}\};$$

  and $\mathcal{F}^v$ is defined analogously.
- For each $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$, the function $f_{\mathcal{S}}\colon \mathcal{V} \to 2^{\mathcal{S}}$ is defined such that $X \in f_{\mathcal{S}}(v)$ iff there is an LTL-structure $(w_i)_{i\geq 0}$ with $w_0 = X$ and $w_i \in \mathcal{S}$, $i \geq 1$, that satisfies the propositional LTL formula $\bigwedge_{f\in\mathcal{F}^v} f$.

For a valuation $v \in \mathcal{V}$, the set $f_{\mathcal{S}}(v)$ thus contains the worlds that may occur in the beginning of some LTL-model (restricted to $\mathcal{S}$) of the *future* subformulas $\mathcal{F}^v$ induced by v. Note that these mappings are independent of the data and hence can be computed in constant time. We can now split the LTL satisfiability test as described below (cf. Lemma 4.3). Here, an LTL-structure *over* $\mathcal{S}$ is such that it uses only worlds from $\mathcal{S}$.

Lemma 6.10. *Let $\mathcal{S} = \{X_1, \dots, X_k\} \subseteq 2^{\{p_1,\dots,p_m\}}$ and $w_0, \dots, w_n \in \mathcal{S}$. The existence of an LTL-structure $\mathfrak{J}$ over $\mathcal{S}$ starting with $w_0, \dots, w_n$ such that $\mathfrak{J}, n \models \phi^{\mathsf{p}}$ is equivalent to the existence of a valuation $v \in \mathcal{V}$ such that*

- *$w_n \in f_{\mathcal{S}}(v)$, and*
- *$(w_0, \dots, w_n, w_n, \dots), n \models \bigwedge_{f \in \mathcal{P}^v} f$.*

Proof. Given such an LTL-structure $\mathfrak{J}$, the valuation v can be obtained by checking which elements of $\{f_1, \dots, f_o\}$ are satisfied at time point n, and the LTL-structure needed for $f_{\mathcal{S}}(v)$ is defined as the substructure of $\mathfrak{J}$ starting at n. Finally, note that the satisfaction of the past formula $\bigwedge_{f \in \mathcal{P}^v}$ in the LTL-structure $(w_0, \dots, w_n, w_n, \dots)$ at time point n does not depend on any time point after n.

Conversely, it is easy to see that $\mathfrak{J}$ can be constructed by appending $w_0, \dots, w_n$ and the LTL-structure obtained from the fact that $w_n \in f_{\mathcal{S}}(v)$, since past and future subformulas are not affected by the worlds after (before) time point n. □

We now focus on the last condition of Lemma 6.10 and hence on the past subformulas. We further consider the following:

- The set $\mathsf{Cl}(\mathcal{P})$ denotes the closure under negation of $\bigcup_{f \in \mathcal{P}} \mathsf{Sub}(f)$.
- The set $\mathcal{Y}$ consists of all *types* for $\mathcal{P}$, which are subsets Y of $\mathsf{Cl}(\mathcal{P})$ that satisfy the following conditions:
  - for every $f \in \mathsf{Cl}(\mathcal{P})$, we have $f \in Y$ iff $\neg f \notin Y$; and
  - for every $f_1 \wedge f_2 \in \mathsf{Cl}(\mathcal{P})$, we have $f_1 \wedge f_2 \in Y$ iff $\{f_1, f_2\} \subseteq Y$.
- For each $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$, the mapping $p_{\mathcal{S}} \colon \mathcal{Y} \to 2^{\mathcal{S}}$ is defined such that $X \in p_{\mathcal{S}}(Y)$ iff
  - whenever $p_i \in Y$, then $p_i \in X$; and
  - whenever $\neg p_i \in Y$, then $p_i \notin X$.
- We call a pair $(Y^{-1}, Y) \in \mathcal{Y} \times \mathcal{Y}$ *t-compatible* if the following hold:
  - $\bigcirc^- f_1 \in Y$ iff $f_1 \in Y^{-1}$;
  - $f_1 \,\mathsf{S}\, f_2 \in Y$ iff either (i) $f_2 \in Y$, or (ii) $f_1 \in Y$ and $f_1 \,\mathsf{S}\, f_2 \in Y^{-1}$.
- A set $Y \in \mathcal{Y}$ is called *initial* if it does not contain formulas of the form $\bigcirc^- f$, and for all $f_1 \,\mathsf{S}\, f_2 \in Y$, we have $f_2 \in Y$.

Lemma 6.11. *Let $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$, $v \in \mathcal{V}$, and $w_0, \dots, w_n \in \mathcal{S}$. Then, we have $(w_0, \dots, w_n, w_n, \dots), n \models \bigwedge_{f \in \mathcal{P}^v} f$ iff there is a mapping $\iota' \colon \{0, \dots, n\} \to \mathcal{Y}$ such that*

- *$\iota'(0)$ is initial and $\mathcal{P}^v \subseteq \iota'(n)$;*
- *for all i, $0 \le i < n$, the pair $(\iota'(i), \iota'(i+1))$ is t-compatible;*
- *for all i, $0 \le i \le n$, we have $w_i \in p_{\mathcal{S}}(\iota'(i))$.*

Proof. The mapping ι' can be defined from $w_0, \dots, w_n$ by considering exactly which past subformulas are satisfied at each time point, which does not depend on time points after n. Obviously, this mapping must be compatible with the worlds w_i (formalized by $p_{\mathcal{S}}$), and satisfy the remaining conditions because of the temporal semantics.

Conversely, given ι', it can be shown by induction on i that for all $f \in \mathsf{Cl}(\mathcal{P})$ we have $f \in \iota'(i)$ iff $(w_0, \dots, w_n, w_n, \dots), i \models f$. The condition on $\iota'(n)$ then yields the claim. □

We now briefly introduce LogTime-bounded ATMs and then apply the above observations to construct an ATM as we need. As in [MBIS90], we assume (deterministic) LogTime TMs to dispose of the following:

- a read-only input tape with an input string of length l,
- a constant number of read/write work tapes, of which the TM can access up to $\mathcal{O}(\log l)$ cells, and
- a read/write input address tape, of which the TM can access up to $\log l$ cells (to hold the address of one input cell).

We thus adopt the random access model of [CKS81], i.e., the symbols on the input tape are accessed by writing the address of the symbol to be read (in binary) on the address tape—instead of by sequentially reading the input. Note that such a model is needed to be able to reach any position of the input string within $\mathcal{O}(\log l)$ time. According to [MBIS90, Lemma 7], a LogTime TM with input of length l can

- add and subtract numbers of $\mathcal{O}(\log l)$ bits and
- determine the logarithm of a binary number of $\mathcal{O}(\log l)$ bits.

We define an *alternating* LogTime Turing Machine with universal and existential states as an extension of a LogTime TM in the usual way (cf. [CKS81]).

We now construct an ATM $\mathcal{M}$ as required for our TCQ satisfiability test. Initially, the two constants sets $\mathcal{S}$ and $\mathcal{B}_\phi$ are guessed (see Lemma 6.8). The idea of the construction is to successively guess the mapping ι' required for Lemma 6.11 as follows. Initially, only the two sets $\iota'(0)$ and $\iota'(n)$ are guessed such that

$\iota'(0)$ is initial and $\mathcal{P}^v \subseteq \iota'(n)$. Note that $\mathcal{Y}$ is constant, and hence this can be done in constant time. The computation proceeds by splitting the sequence $0, \dots, n$ in half, guessing a new pair $(\iota'(\frac{n+1}{2}), \iota'(\frac{n+1}{2}+1))$[12] and checking it for t-compatibility, and proceeding with two copies of the ATM who remember only the pairs $(\iota'(0), \iota'(\frac{n+1}{2}))$ and $(\iota'(\frac{n+1}{2}+1), \iota'(n))$, respectively. These copies are now responsible for checking the conditions of Lemma 6.11 for the two subsequences $0, \dots, \frac{n+1}{2}$ and $\frac{n+1}{2}+1, \dots, n$, respectively. Theses sequences are successively split in this manner until each copy knows two sets $\iota'(i)$, $\iota'(i+1)$ for adjacent time points i and $i+1$, which are then also checked for t-compatibility. Note that in this way we only guess each $\iota'(i)$ once, i.e., no conflicting guesses for the same time point occur. It remains to check for each i, $0 \leq i \leq n$, whether is a $w_i \in \mathcal{S}$ such that $w_i \in p_{\mathcal{S}}(\iota'(i))$, $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},w_i}(i)$, and $\mathsf{DB} \not\models \mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},w_i,\mathcal{B}_\phi)}(i)$ for any $S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O}) \setminus \mathsf{N}_\mathsf{RR}^-$ and $a \in \mathsf{N}_\mathsf{I}(\phi)$ with $\exists S(a) \notin \mathcal{B}_\phi$ (cf. Lemma 6.8). Additionally, we must ensure that $w_n \in f_{\mathcal{S}}(v)$ (cf. Lemma 6.10). An special copy of the ATM verifies that $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},X}(-1)$ holds for all $X \in \mathcal{S}$. For the remaining part of Lemma 6.8, we need to guess, for each $\exists S(a) \in \mathcal{B}_\phi$ of the form above, which time point i satisfies $\mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},w_i,\mathcal{B}_\phi)}(i)$. This can be done by maintaining in each branch of the computation the information which elements of $\mathcal{B}_\phi$ it is responsible for. If all these checks succeed, the guessed sets $\iota'(0), \dots, \iota'(n)$ obviously satisfy Lemma 6.11, and we can define $\iota(i)$ in such a way that $X_{\iota(i)} = w_i$ (where $\mathcal{S} = \{X_1, \dots, X_k\}$), in order to satisfy Lemmas 4.3 and 6.8. The behavior of the ATM is illustrated in Figure 1.

For the tests of the form $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},X}(i)$ and $\mathsf{DB} \models \mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}(i)$, we employ ATMs $\mathcal{M}_{\mathsf{rsat}_{\mathcal{S},X}}$ and $\mathcal{M}_{\mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}}$, respectively, that run in logarithmic time. Such machines exist since the former problem is in AC^0 [CDGL+05, CDGL+09], which is contained in ALogTime. These ATMs, which are used only at the end of each computation path of our ATM $\mathcal{M}$, receive as input the index i of the current time point, and the constant sets $\mathcal{S}$, $\mathcal{B}_\phi$, $w_i := X$ (which is guessed for each i independently), and possibly a constant assertion $\exists S(a)$.

Note that the number n can be retrieved from the input via an FO-query, which is why we assume it to be given with the input (e.g., a database could provide the number n in a view defined by the FO(<)-query $\neg\exists t.t > n$). We hence can assume n to be written on the input tape in binary, at the beginning of the tape, separated from the other input by a special marker symbol. The rest of the input consists simply of the database DB, in whatever format is required by the machines $\mathcal{M}_{\mathsf{rsat}_{\mathcal{S},X}}$ and $\mathcal{M}_{\mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}}$. Recall that all other parts of the input, in particular m and $\mathcal{O}$, are constant, and hence can be encoded into the ATM itself.

Next to the input and address tape, $\mathcal{M}$ has several work tapes to store the following information:

[12]For ease of presentation, we will assume in the following that $n+1$ is a power of 2. If this is not the case, the ATM would have to handle non-uniform divisions of the sequence $0, \dots, n$.

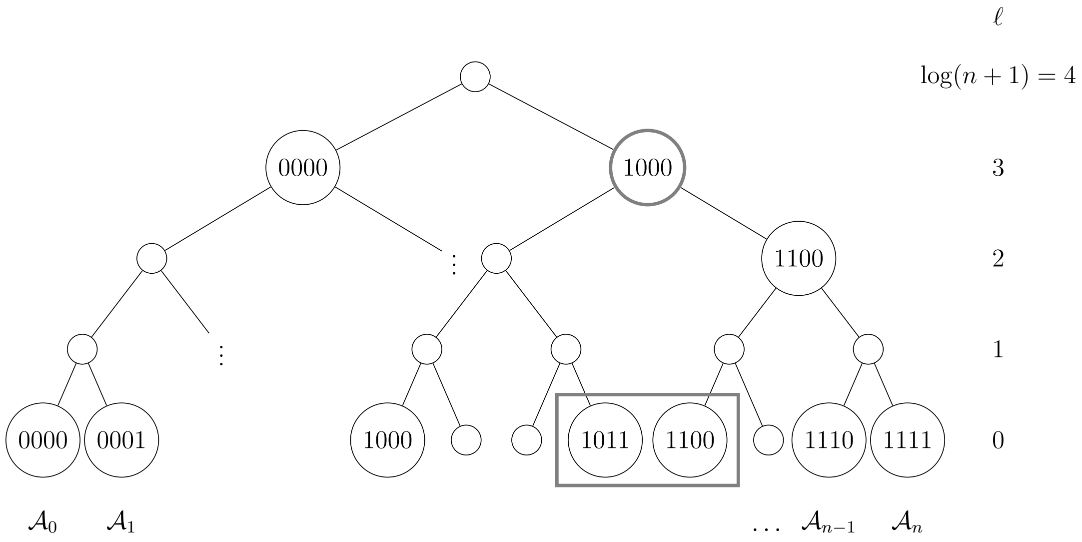

Figure 1: A sketch of the computation of the ATM for $n = 15$. The number ℓ denotes the current level of the computation tree. The nodes are labeled with the index i (in binary notation) that represents the computation path by designating the left border of the currently considered subsequence of $0, \dots, n$. The copy of the ATM designated by the marked node ($i = 1000$, $\ell = 3$) guesses a t-compatible pair of sets from $\mathcal{Y}$, which corresponds to $(\iota'(1011), \iota'(1100))$. The ATM then splits into two copies for each subtree, each continuing with one of the guessed indices.

1. the sets $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$ and $\mathcal{B}_\phi \subseteq \{B(a) \mid B \in \mathsf{BC}(\mathcal{O}),\ a \in \mathsf{N_I}(\phi)\}$ considered by the ATM, and a subset $\mathcal{B}'_\phi \subseteq \{\exists S(a) \in \mathcal{B}_\phi \mid S \in \mathsf{N}^-_\mathsf{R}(\mathcal{O}) \setminus \mathsf{N}^-_\mathsf{RR}\}$; temporarily, one additional such subset may be stored on this tape;
2. the currently considered level ℓ of the computation tree;
3. the currently considered branch of the tree, represented by the index i of the left border of the considered subsequence of $0, \dots, n$;
4. two sets $Y_l, Y_r \in \mathcal{Y}$, which are associated with the left and right border, respectively, of the currently considered subsequence of $0, \dots, n$; temporarily, two additional sets from $\mathcal{Y}$ are stored on this tape;
5. the valuation $v \in \mathcal{V}$ considered by the ATM; and
6. additional work tapes needed for $\mathcal{M}_{\mathsf{rsat}_{\mathcal{S},X}}$.

Thus, we have a constant number of read/write work tapes. Moreover, the total length of tapes 1, 4, and 5 is constant, tape 2 requires $\mathcal{O}(\log\log(n+1))$ bits, and tape 3 requires $\log(n + 1)$ bits.

$\mathcal{M}$ works as follows:

1. $\mathcal{M}$ guesses sets $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_m\}}$ and $\mathcal{B}_\phi \subseteq \{B(a) \mid B \in \mathsf{BC}(\mathcal{O}),\ a \in \mathsf{N_I}(\phi)\}$, computes $\mathcal{B}'_\phi := \{\exists S(a) \in \mathcal{B}_\phi \mid S \in \mathsf{N}^-_\mathsf{R}(\mathcal{O}) \setminus \mathsf{N}^-_\mathsf{RR}\}$, and stores all three sets on tape 1. Then, $\mathcal{M}$ guesses a valuation $v \in \mathcal{V}$ and stores it on tape 5.

2. The ℓ-counter on tape 2 is initialized to $\ell := \log(n + 1)$.

3. The index of the left border stored on tape 3 is initialized to $i := 0$.

4. $\mathcal{M}$ guesses two sets $\iota'(0)$ and $\iota'(n)$ from $\mathcal{Y}$, checks whether $\iota'(0)$ is initial and whether $\mathcal{P}^v \subseteq \iota'(n)$, and stores the sets on tape 4.

5. $\mathcal{M}$ then continuously executes the below steps, while $\ell > 1$:
   (a) $\mathcal{M}$ decreases ℓ by 1. Hence, the sets $\iota'(i)$ and $\iota'(i + 2^{\ell+1} - 1)$ are currently stored on tape 4.[13]
   (b) $\mathcal{M}$ guesses two new sets $\iota'(i + 2^\ell - 1)$ and $\iota'(i + 2^\ell)$ and stores them temporarily on tape 4.
   (c) $\mathcal{M}$ checks whether the pair $(\iota'(i + 2^\ell - 1), \iota'(i + 2^\ell))$ is t-compatible. This is a simple syntactic check on two sets of constant size.
   (d) $\mathcal{M}$ guesses a partition of $\mathcal{B}'_\phi$ into $\mathcal{B}^1_\phi$ and $\mathcal{B}^2_\phi$.
   (e) $\mathcal{M}$ splits into two copies. The "left" copy continues the computation with the sets $\iota'(i)$ and $\iota'(i + 2^\ell - 1)$ on tape 4, the index i on tape 3, and $\mathcal{B}'_\phi := \mathcal{B}^1_\phi$ on tape 1. The "right" copy continues with $\iota'(i + 2^\ell)$, $\iota'(i + 2^{\ell+1} - 1)$, $i + 2^\ell$, and $\mathcal{B}^2_\phi$. Hence, they are responsible for the subsequences $i, \dots, i + 2^\ell - 1$ and $i + 2^\ell, \dots, i + 2^{\ell+1} - 1$, respectively.

6. When ℓ reaches 1, this counter is decreased one final time. Tape 4 now holds the adjacent sets $\iota'(i)$ and $\iota'(i + 1)$. $\mathcal{M}$ then checks whether $(\iota'(i), \iota'(i + 1))$ is t-compatible. $\mathcal{M}$ again guesses a partition of $\mathcal{B}_\phi$ into $\mathcal{B}^{(i)}_\phi$ and $\mathcal{B}^{(i+1)}_\phi$.

7. Then, $\mathcal{M}$ again splits into two copies. The "left" copy now holds only the single set $\iota'(i)$ on tape 4, i on tape 3, and $\mathcal{B}^{(i)}_\phi$ on tape 1.. The "right" copy continues with $\iota'(i + 1)$, $i + 1$, and $\mathcal{B}^{(i+1)}_\phi$.

   Additionally, at $i = 0$ a special third copy is created where tape 4 is empty and tape 3 contains -1.

8. In the copies with $\ell = 0$ and $i \in \{0, \dots, n\}$, we know that $\iota'(i)$ is stored on tape 4 and $\mathcal{B}^{(i)}_\phi$ is stored on tape 1. $\mathcal{M}$ now guesses a $w_i \in \mathcal{S}$ and verifies the following conditions:
   - $w_i \in p_\mathcal{S}(\iota'(i))$;
   - if $i = n$, then $w_n \in f_\mathcal{S}(v)$;
   - $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},w_i}(i)$ (using $\mathcal{M}_{\mathsf{rsat}_{\mathcal{S},w_i}}$ with additional input i);

[13] In the first iteration, we have $i = 0$ and $i + 2^{\ell+1} - 1 = 2^{\log(n+1)} - 1 = n$.

- $\mathsf{DB} \not\models \mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},w_i,\mathcal{B}_\phi)}(i)$ for all $S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O}) \setminus \mathsf{N}_\mathsf{RR}^-$ and $a \in \mathsf{N}_\mathsf{I}(\phi)$ with $\exists S(a) \notin \mathcal{B}_\phi$ (using $\mathcal{M}_{\mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},X,\mathcal{B}_\phi)}}$); and
- $\mathsf{DB} \models \mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},w_i,\mathcal{B}_\phi)}(i)$ for all $\exists S(a) \in \mathcal{B}_\phi^{(i)}$.

9. In the remaining copy with $\ell = 0$ and $i = -1$, $\mathcal{M}$ splits into one copy for each $X \in \mathcal{S}$, and checks whether $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},X}(-1)$ (using $\mathcal{M}_{\mathsf{rsat}_{\mathcal{S},X}}$ with input -1).

If any of the described tests fail, the current copy of $\mathcal{M}$ halts in a rejecting state, which results in an unsuccessful run. Otherwise, each copy of $\mathcal{M}$ halts in an accepting state, confirming the satisfiability of ϕ w.r.t. $\mathcal{K}$.

Theorem 6.12. *Entailment of r-simple TCQs in DL-Lite$^{\mathcal{H}}_{horn}$ is in* ALogTime *w.r.t. data complexity, even if* $\mathsf{N}_\mathsf{RR} \neq \emptyset$.

Proof. Let ϕ be an r-simple TCQ, $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0 \le i \le n} \rangle$ be a TKB, and DB be defined as above. Note that DB is simply a different representation of the ABox sequence $(\mathcal{A}_i)_{0 \le i \le n}$ that is of the same size. The ATM $\mathcal{M}$ accepts the input n and DB (in logarithmic time) iff there are $\mathcal{S} = \{X_1, \dots, X_k\} \subseteq 2^{\{p_1,\dots,p_m\}}$, $\mathcal{B}_\phi \subseteq \{B(a) \mid B \in \mathsf{BC}(\mathcal{O}), a \in \mathsf{N}_\mathsf{I}(\phi)\}$, $v \in \mathcal{V}$, $\iota' \colon \{0, \dots, n\} \to \mathcal{Y}$, and worlds $w_0, \dots, w_n \in \mathcal{S}$ such that

- $\iota'(0)$ is initial and $\mathcal{P}^v \subseteq \iota'(n)$;
- for each i, $0 \le i < n$, the pair $(\iota'(i), \iota'(i+1))$ is t-compatible;
- for each i, $0 \le i \le n$, we have $w_i \in p_\mathcal{S}(\iota'(i))$;
- $w_n \in f_\mathcal{S}(v)$;
- for each i, $0 \le i \le n$, we have $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},w_i}(i)$;
- for each $S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O}) \setminus \mathsf{N}_\mathsf{RR}^-$ and $a \in \mathsf{N}_\mathsf{I}(\phi)$, we have $\exists S(a) \in \mathcal{B}_\phi$ iff there exists an i, $0 \le i \le n$, with $\mathsf{DB} \models \mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},w_i,\mathcal{B}_\phi)}(i)$; and
- for each $X \in \mathcal{S}$ it holds that $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},X}(-1)$.

By Lemmas 6.10 and 6.11, this is equivalent to the existence of $\mathcal{S}$ and w_i as above and an LTL-structure $\mathfrak{J}$ such that

- $\mathfrak{J}$ starts with $w_0, \dots, w_n$;
- $\mathfrak{J}$ uses only worlds from $\mathcal{S}$;
- $\mathfrak{J}, n \models \phi^\mathsf{p}$;

- for each i, $0 \leq i \leq n$, we have $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},w_i}(i)$;
- for each $S \in \mathsf{N}_\mathsf{R}^-(\mathcal{O}) \setminus \mathsf{N}_\mathsf{RR}^-$ and $a \in \mathsf{N}_\mathsf{I}(\phi)$, we have $\exists S(a) \in \mathcal{B}_\phi$ iff there exists an i, $0 \leq i \leq n$, with $\mathsf{DB} \models \mathsf{pr}_{(\exists S(a),\mathcal{O}|\mathcal{S},w_i,\mathcal{B}_\phi)}(i)$; and
- for each $X \in \mathcal{S}$ it holds that $\mathsf{DB} \models \mathsf{rsat}_{\mathcal{S},X}(-1)$.

Due to the condition that each w_i is an element of $\mathcal{S}$, the sequence $w_0, \ldots, w_n$ can equivalently be expressed by a mapping $\iota\colon \{0,\ldots,n\} \to \{1,\ldots,k\}$ with $w_i = X_{\iota(i)}$ for all i, $0 \leq i \leq n$.

Hence, by Lemmas 6.8 and 6.9, the above is equivalent to the existence of $\mathcal{S}$ and ι and an r-complete tuple w.r.t. $\mathcal{S}$ and ι such that ϕ^{p} is t-satisfiable w.r.t. $\mathcal{S}$ and ι. Finally, by Lemmas 4.3 and 4.9, we obtain the equivalence to the satisfiability of ϕ w.r.t. $\mathcal{K}$. The claim follows from the fact that the class ALogTime is closed under complement (see [CKS81, Theorem 2.5]). □

7 Beyond the *horn* Fragment

In this section, we show that for the *krom* and *bool* fragments of *DL-Lite*, the above complexity results do not apply any more, even if role hierarchies are omitted. In particular, TCQ entailment gets as hard as for very expressive DLs, such as $\mathcal{SHQ}$.

7.1 Lower Bounds

For data complexity, the co-NP lower bound follows from co-NP-hardness of conjunctive query answering w.r.t. *DL-Lite*$_{krom}$-knowledge bases. The latter is a consequence of [CDL+07, Theorem 48 (1)], where the hardness is stated for *DL-Lite*$_{core}$ extended by CIs that allow for $\neg A$, $A \in \mathsf{N}_\mathsf{C}$, on the left-hand side. In the remainder of this section, we hence focus on combined complexity and investigate lower bounds of TCQ entailment in *DL-Lite*$_{krom}$.

Our query formalism enables us to express several kinds of GCIs not expressible in *DL-Lite*$_{krom}$ via appropriate negated CQs (cf. Table 7.1).[14] We use (fresh) symbols of the form $\overline{A}_1$ to denote the complements of given concept names A_1, which can be expressed by the CIs described in the following lemma.

Lemma 7.2. *Let* $(C \sqsubseteq D, \phi)$ *be one of the pairs of a GCI and a TCQ given in Table 7.1 and* $\mathcal{I}$ *be a model of* $\top \sqsubseteq A_i \sqcup \overline{A}_i$ *and* $A_i \sqcap \overline{A}_i \sqsubseteq \bot$, *for all concept names* A_i *occurring in* D. *Then, we have* $\mathcal{I} \models C \sqsubseteq D$ *iff* $\mathcal{I} \models \phi$.

[14] We assume the reader to be familiar with the common semantics of these GCIs. A good introduction is given in [BCM+03].

GCI	TCQ
$\exists R.A_1 \sqsubseteq A_2$	$\neg\exists x, y.R(x,y) \wedge A_1(y) \wedge \overline{A}_2(x)$
$A_1 \sqsubseteq \forall R.A_2$	$\neg\exists x, y.A_1(x) \wedge R(x,y) \wedge \overline{A}_2(y)$
$A_1 \sqcap \cdots \sqcap A_m \sqsubseteq A_{m+1} \sqcup \cdots \sqcup A_{m+n}$	$\neg\exists x.A_1(x) \wedge \cdots \wedge A_m(x) \wedge \overline{A}_{m+1}(x) \wedge \ldots \overline{A}_{m+n}(x)$

Table 7.1: The rules of our transformation

Proof. Note that ϕ is of the form $\neg\psi$ with ψ being a CQ. We now assume $\mathcal{I} \not\models \neg\psi$ and thus get $\mathcal{I} \models \psi$, and a corresponding homomorphism, by Definition 2.5. Especially note that the atoms in the CQ ψ always refer to the concepts and roles of the corresponding GCI $C \sqsubseteq D$ in the same way (i.e., C and $\neg D$ are modeled in the CQ). We thus have an element e in the domain of $\mathcal{I}$ such that $e \in C^{\mathcal{I}}$ and $e \notin D^{\mathcal{I}}$, by our assumption on the GCIs w.r.t. $\overline{D}$ in $\mathcal{O}$ and the semantics of the $\forall$ constructor. This directly yields $C^{\mathcal{I}} \not\subseteq D^{\mathcal{I}}$ and thus $\mathcal{I} \not\models C \sqsubseteq D$. The other direction follows from similar arguments. □

We thus can always use GCIs as described above in the ontology we construct for proving hardness of TCQ entailment. More precisely, by the above lemma, we have $\langle\mathcal{O}, (\mathcal{A}_i)_{0\leq i\leq n}\rangle \models \phi$ iff $\langle\mathcal{O}', (\mathcal{A}_i)_{0\leq i\leq n}\rangle \models ((\Box\Box^-\psi) \rightarrow \phi)$, where $\mathcal{O}'$ is obtained by removing all GCIs of the forms listed in Table 7.1 from $\mathcal{O}$ and adding the necessary CIs to express the complements $\overline{A}_i$, and ψ is the conjunction of the negated CQs simulating the removed GCIs. With the same construction, ϕ is satisfiable w.r.t. $\langle\mathcal{O}, (\mathcal{A}_i)_{0\leq i\leq n}\rangle$ iff $(\Box\Box^-\psi) \wedge \phi$ is satisfiable w.r.t. $\langle\mathcal{O}', (\mathcal{A}_i)_{0\leq i\leq n}\rangle$. This means that we can also use CIs of *DL-Lite$_{bool}$* in *DL-Lite$_{krom}$*, which yields the following corollary.

Corollary 7.3. *TCQ entailment in DL-Lite$_{bool}$ can be polynomially reduced to TCQ entailment in DL-Lite$_{krom}$.*

This enables us to directly derive two rather strong lower bounds, even without any rigid symbols. They follow from ExpTime-hardness of UCQ entailment in *DL-Lite$_{bool}$* [BMP14, Corollary 2] and 2-ExpTime-hardness of UCQ entailment in *DL-Lite$^{\mathcal{H}}_{bool}$* [BMP13, Theorem 12].

Theorem 7.4. *Regarding combined complexity, TCQ entailment is*

- ExpTime-*hard in DL-Lite$_{krom}$ and*
- 2-ExpTime-*hard in DL-Lite$^{\mathcal{H}}_{krom}$,*

even if $\mathsf{N_{RC}} = \emptyset$ *and* $\mathsf{N_{RR}} = \emptyset$.

In addition to the CIs allowed in *DL-Lite$_{bool}$*, more complex GCIs with nested conjunctions and disjunctions can be reduced to the forms in Table 7.1 by introducing appropriate abbreviations, as long as $\forall R.A$ only appears on the right-hand side and $\exists R.A$ only appears on the left-hand side. For example, the GCI

$$A_1 \sqcup A_2 \sqcup \exists R.A_3 \sqsubseteq A_4 \sqcup \forall R.(A_1 \sqcap \exists S)$$

can be expressed by

$$A_1 \sqsubseteq A_4 \sqcup A',\ A_2 \sqsubseteq A_4 \sqcup A',\ A''' \sqsubseteq A_4 \sqcup A',$$
$$\exists R.A_3 \sqsubseteq A''',\ A' \sqsubseteq \forall R.A'',\ A'' \sqsubseteq A_1,\ A'' \sqsubseteq \exists S.$$

These GCIs can then again be simulated by negated CQs as described above. Moreover, this transformation is always polynomial.

In the following, we provide further reductions to TCQ entailment in *DL-Lite*$_{krom}$ (i.e., for the two cases of considered rigid symbols). We therein use such complex GCIs without further notice.

For the case where $\mathsf{N_{RR}} = \emptyset$ but possibly $\mathsf{N_{RC}} \neq \emptyset$, we apply a result of [BT15c, BT15a], where NExpTime-hardness is shown for the satisfiability problem of formulas in $\mathcal{EL}_\bot$-LTL, a formalism similar to TCQs, in the case that $\mathsf{N_{RC}} \neq \emptyset$ but $\mathsf{N_{RR}} = \emptyset$. Formulas in $\mathcal{EL}_\bot$-LTL are similar to TCQs in that they consist of $\mathcal{EL}_\bot$-axioms (i.e., assertions and GCIs which may contain the concept constructors $\top$, $\bot$, $\sqcap$, and qualified existential restrictions) that are combined via the LTL operators. The corresponding satisfiability problem further differs from TCQ entailment because neither a global ontology nor a sequence of ABoxes is considered.

Theorem 7.5. *TCQ entailment in DL-Lite*$_{krom}$ *is* co-NExpTime*-hard w.r.t. combined complexity if* $\mathsf{N_{RC}} \neq \emptyset$*.*

Proof. The proof in [BT15a] is a reduction from a NExpTime-hard variant of the domino problem. However, the assertions used in the $\mathcal{EL}_\bot$-LTL formula ϕ constructed there are of the form $A(a)$ for $A \in \mathsf{N_C}$ and $a \in \mathsf{N_I}$, and hence can already be seen as TCQs. Furthermore, all GCIs occurring in ϕ are of the form $\top \sqsubseteq A_1$, $A_1 \sqsubseteq \bot$, or $A_1 \sqcap A_2 \sqsubseteq A_3$, and hence by Lemma 7.2 we can directly replace them by negated CQs according to Table 7.1, without affecting the semantics. □

The next theorem covers the remaining case where $\mathsf{N_{RR}} \neq \emptyset$.

Theorem 7.6. *TCQ entailment in DL-Lite*$_{krom}$ *is* 2-ExpTime*-hard w.r.t. combined complexity if* $\mathsf{N_{RR}} \neq \emptyset$*.*

Proof. For the proof, we adapt a reduction proposed in [BGL12] by using ideas of [KRH13]. [BGL12] reduce the word problem for exponentially space bounded alternating Turing machines to the satisfiability problem in $\mathcal{ALC}_{|gGCI}$-LTL (a formalism similar to $\mathcal{EL}_\bot$-LTL).

We first provide some details on the ATMs we consider. An ATM is a tuple $\mathcal{M} = (\mathcal{Q}, \Sigma, \Gamma, q_0, \Delta)$, where

- $\mathcal{Q} = \mathcal{Q}_\exists \cup \mathcal{Q}_\forall \cup \{q_a, q_r\}$ is a finite set of states, partitioned into *existential states* ($\mathcal{Q}_\exists$), *universal states* ($\mathcal{Q}_\forall$), an *accepting state* q_a, and a *rejecting state* q_r;
- Σ is the *input alphabet* with $\Sigma \subseteq \Gamma$;
- Γ is the set of *working symbols* containing a *blank symbol* $B \notin \Sigma$;
- $q_0 \in \mathcal{Q}_\exists \cup \mathcal{Q}_\forall$ is the *initial state*; and
- Δ denotes the *transition relation*, for which we have

  $$\Delta \subseteq \mathcal{Q} \times \Gamma \times \mathcal{Q} \times \Gamma \times \{L, R\}.$$

  We use $\Delta(q, \sigma)$ to denote the set $\{(p, \varrho, M) \mid (q, \sigma, p, \varrho, M) \in \Delta\}$.

As usual, the computation of an ATM on an input word is described as a sequence of *configurations*, and we speak of a *halting* configuration if the ATM is in an accepting or rejecting state. For both the different kinds of state and the transition relation we employ the usual semantics [CKS81]. We assume w.l.o.g. that an ATM never moves to the left when it is on the left-most tape cell, that any configuration which is no halting configuration has at least one successor configuration, and that all computations of an ATM are finite (cf. [CKS81, Theorem 2.6]). Further, we may assume that the length of every computation on a word $w \in \Sigma^k$ is bounded by 2^{2^k}, and that every configuration in such a computation can be represented using $\leq 2^k$ symbols, plus one to represent the state.

Given an ATM $\mathcal{M}$ and an input word w, the decision problem we focus on is the *word problem*: the question if $\mathcal{M}$ accepts w or not. According to [CKS81, Corollary 3.5], there is an exponentially space-bounded ATM $\mathcal{M} = (\mathcal{Q}, \Sigma, \Gamma, q_0, \Delta)$ with only finite computations, for which the word problem is 2-ExpTime-hard. In what follows, we show that this problem can be reduced to TCQ satisfiability in *DL-Lite*$_{krom}$ with rigid role names.

To this end, let $w = \sigma_0 \ldots \sigma_{k-1} \in \Sigma^*$ be an arbitrary input word given to $\mathcal{M}$. We next construct a TCQ $\phi_{\mathcal{M},w}$ and a TKB $\langle \mathcal{O}_{\mathcal{M},w}, (\mathcal{A}_0) \rangle$ in *DL-Lite*$_{krom}$ such that $\mathcal{M}$ accepts w iff $\phi_{\mathcal{M},w}$ is satisfiable w.r.t. $\langle \mathcal{O}_{\mathcal{M},w}, (\mathcal{A}_0) \rangle$.

To get an intuition of the reduction, consider Figure 2, which shows (parts of) example computations of such an ATM. In the tree describing all computations (i.e., one path describes one computation), the individual configurations are represented explicitly, one after the other, and each as a chain, such that every tree node represents one of the 2^k tape cells of a configuration, which are numbered by the A counter. Each tree node or cell is then represented by an individual in the reduction; and, since these individuals are connected by rigid roles, the computation tree 'exists' at all time points, numbered by the A' counter. We now exemplarily describe the modeling of the transitions and corresponding successor

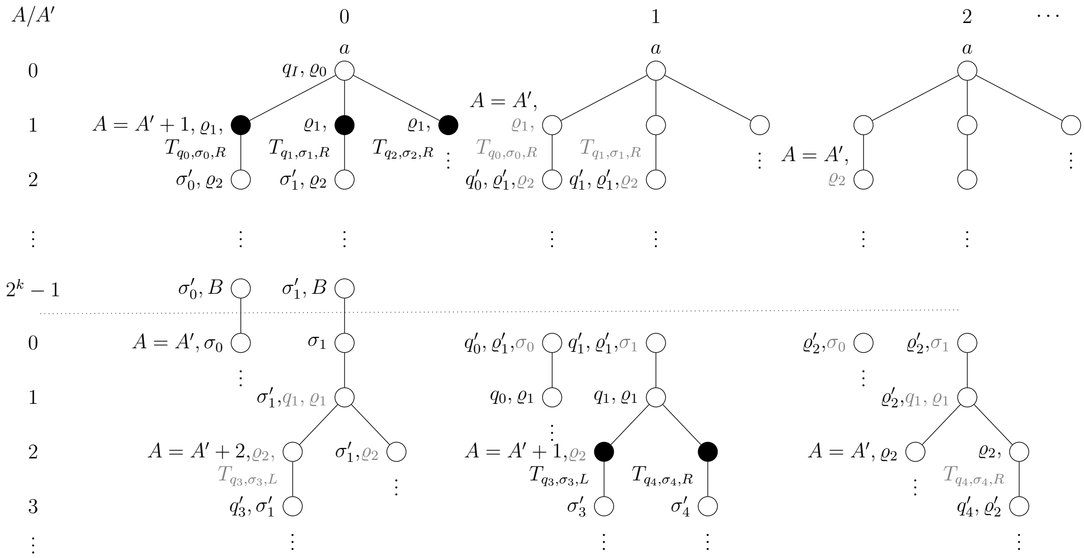

Figure 2: A sketch of the modeling of an exemplary computation tree of the ATM. The tree nodes represent domain individuals and are labeled with relevant concepts. The named individual a represents the first cell in the initial configuration, q_I is the initial machine state, and ϱ_0 the symbol in the first tape cell. Some of the rigid concepts are printed in gray to differentiate the time points where they are induced. The figure also abstracts from the fact that the temporal counter A' has to be considered modulo 2^k if used in concepts such as $A = A'$.

configurations. In particular, we 'propagate' the new state q and symbol σ of a transition and the cell contents ϱ (of the considered configuration) that do not change to the successor configuration, which follows in the tree. To represent all this information, we use corresponding *rigid* concept names and then propagate them via *flexible* concept names (marked by a prime) using the temporal dimension. More specifically, the tape contents of cell i are propagated to the successor configuration if the individual that represents the cell satisfies $A = A'$ (e.g., ϱ_1 and ϱ_2 at time points 1 and 2, respectively) and the cell is not under the head (e.g., ϱ_0 is not propagated). The cell under the head can be identified because it satisfies a state symbol q. It thus determines the state of the considered configuration and, together with the symbol in the cell, the transitions to be considered. In a universal state, for each such transition, the configurations have one branch identified by a specific concept of the form $T_{q_0,\sigma_0,M}$ (for an existential state, there is only one branch). The concept $T_{q_0,\sigma_0,M}$ initiates the propagation of the symbol σ_0 to the successor configuration if $A = A'+1$. In this way, the propagation stops correctly in the next configuration at the cell that has been previously under the head, which satisfies $A = A'$ (i.e., it is left to the cell that previously satisfied $T_{q_0,\sigma_0,M}$). The new state q_0 is similarly propagated, but the corresponding time point specifically depends on the direction of the move M. For a right move, the propagation happens if the individual satisfying $T_{q_0,\sigma_0,M}$ also satisfies $A = A'$

and thus stops at the same cell in the successor configuration (recall that the individual satisfying $T_{q_0,\sigma_0,M}$ represents the cell right to the head) (e.g., see the propagation of q_0 and q_1 at time point 1). For left moves, the propagation should stop two cells left to the individual satisfying a concept of the form $T_{q_3,\sigma_3,L}$; we thus require the individual to satisfy $A = A' + 2$ to start the propagation.

Before formally describing this intuition, we now introduce all symbols we use in detail:

- A single named individual a identifies the root of the tree.
- Rigid role names $R_{q,\varrho,M}$, for all $q \in \mathcal{Q}$, $\varrho \in \Gamma$, $M \in \{L, R\}$, represent the edges of the tree. We collect all these role names into the set $\mathcal{R}$.

  Note that these roles represent the major difference to the reduction of [BGL12], where a single rigid role fulfills this purpose, but is used within qualified existential restrictions on the right-hand side of GCIs.
- A rigid concept name, for each element in $\mathcal{Q} \cup \Gamma$, represents the tape content, the current state, and the head position in each configuration in the tree: if $\mathcal{M}$ is in a state q and the head is on the i-th tape cell, then the individual (tree node) representing this cell satisfies the concept name q, and correspondingly for the symbols in Γ.
- Rigid concept names $A_0, \ldots, A_{k-1}$, are used to model the bits of a binary counter numbering the tape cells in the configurations.
- Rigid concept names I and H point out special cells. In particular, I is satisfied by the nodes representing the initial configuration, and H is satisfied by all nodes representing a tape cell that is located (anywhere) to the right of the head in the current configuration.
- The rigid concept names $T_{q,\varrho,M}$, for all $q \in \mathcal{Q}$, $\varrho \in \Gamma$, $M \in \{L, R\}$, are satisfied by an individual, representing a cell, if the head is on the left neighboring cell and the ATM executes the transition (q, σ, M), in the described configuration.

We use the temporal dimension to synchronize successor configurations in accordance with the chosen transition, i.e., to model the change in tape contents, head position, and state from one configuration to the next:

- Flexible concept names $A'_0, \ldots, A'_{k-1}$ are used to model a counter in the temporal dimension. Dual to the counter $A_0, \ldots, A_{k-1}$, its value is incremented (modulo 2^k) along time but, at every time point, all individuals share the value of this counter. It is used for the synchronization of successor configurations: if the A' counter has value i, then the symbol in the i-th tape cell of any configuration (where i is not the head position) is propagated to the i-th tape cell of its successor configuration. Similarly, the state is propagated from the cells c directly right of the head position, each pointing out a specific transition (via the symbols $T_{q,\varrho,M}$), to the corresponding cells of the successor configurations (i.e., these cells have the same position on the tape as c for right-moves and otherwise lie two to the left);

- We further use a flexible concept name, for each element in $\mathcal{Q} \cup \Gamma$, similarly distinguished from the rigid version by a prime. Considering a fixed time point, these names are used for the propagation of the state q or cell content σ of a cell c to the corresponding cell in the successor configuration(s). This propagation happens via the right neighboring cells of that configuration, which then satisfy q' and σ', respectively, at the time point whose A'-counter corresponds to the A-counter at c.

We may further use concept names of the form $\overline{A}$, for given concept names A, as detailed in Lemma 7.2.

We now define the TCQ $\phi_{\mathcal{M},w}$ and the TKB $\langle \mathcal{O}_{\mathcal{M},w}, (\mathcal{A}_0) \rangle$, by describing the conjuncts of $\phi_{\mathcal{M},w}$ and listing the GCIs contained in $\mathcal{O}_{\mathcal{M},w}$. To enhance readability, we may use GCIs that are not in *DL-Lite*$_{krom}$, but can be transformed as described in the beginning of this section. We first express the tree structure in general:

- We enforce all elements to have some successor except if they satisfy q_a or q_r. Since the only elements satisfying a symbol from $\mathcal{Q}$ are the ones representing the position of the head, the tree generation thus is only stopped if we meet a halting configuration:

$$\overline{q_a} \sqcap \overline{q_r} \sqsubseteq \bigsqcup_{R_\delta \in \mathcal{R}} \exists R_\delta.$$

  We use a big disjunction over all possible roles since we do not know which transition will be chosen, yet.

- The A-counter is initialized with value 0 at a and incremented alongside the tree, modulo 2^k. Thus, all elements representing the first tape cell in some configuration in the tree satisfy the auxiliary concept name $C_{A=0}$ that is defined by

$$C_{A=0} \equiv \overline{A}_0 \sqcap \ldots \sqcap \overline{A}_{k-1}.$$

  In what follows, we use additional concept names of the form $C_{A=i}$, for polynomially many values i, which are defined similarly.

  We further add the conjunct

$$C_{A=0}(a)$$

to $\mathcal{A}_0$. Note that $\overline{A}_0, \ldots, \overline{A}_{k-1}$ are rigid, and hence this assertion must be satisfied at every time point. We use the following GCIs to model the counter, for all i, $0 \leq i \leq k-1$:

$$\begin{aligned}
\bigsqcap_{0 \leq j \leq i} A_j &\sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.\overline{A}_i, \\
\bigsqcap_{0 \leq j < i} A_j \sqcap \overline{A}_i &\sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.A_i, \\
\Big(\bigsqcup_{0 \leq j < i} \overline{A}_j\Big) \sqcap A_i &\sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.A_i, \\
\Big(\bigsqcup_{0 \leq j < i} \overline{A}_j\Big) \sqcap \overline{A}_i &\sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.\overline{A}_i.
\end{aligned}$$

For example, if the bits A_0, ..., A_i are all true in the current tape cell, then in the successor cell these bits are all false.

We thus have described a sequence of configurations, where we can address single tape cells in all the configurations using the A-counter, which restarts every time it has reached $2^k - 1$, i.e., after every representation of a configuration. We now enforce basic conditions which help to ensure that the tree actually represents an accepting computation of $\mathcal{M}$ on w.

- To formulate these conditions, we use the rigid concept name H to identify the tape cells that are to the right of the head:

  $$\Big(H \sqcup \bigsqcup_{q \in \mathcal{Q}} q\Big) \sqcap \overline{C}_{A=2^k-1} \sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.H.$$

  Especially note that the propagation stops at tree levels whose elements represent the last cell in a configuration, since these elements satisfy $C_{A=2^k-1}$.

- There is only one head position per configuration:

  $$H \sqsubseteq \bigsqcap_{q \in \mathcal{Q}} \overline{q}.$$

  Note that we do not have to consider the elements representing the cells left to the head since, if such a cell satisfies a concept name from $\mathcal{Q}$, then all its successors in the tree are enforced to satisfy H.

- Each tape cell is associated with at most one state (which, at the same time, represents the position of the head):

  $$\top \sqsubseteq \bigsqcap_{q_1, q_2 \in \mathcal{Q}, q_1 \neq q_2} \overline{q_1} \sqcup \overline{q_2}$$

- Each tape cell contains exactly one symbol:

$$\top \sqsubseteq \bigsqcup_{\sigma\in\Gamma}\Big(\sigma \sqcap \bigsqcap_{\sigma'\in\Gamma\setminus\{\sigma\}} \overline{\sigma'}\Big),$$

Before specifying the remaining, more intricate conditions for the synchronization of the configurations, we now describe the first configuration in the tree (starting at a) as the initial configuration.

- In particular, we mark the corresponding elements by adding the assertion $I(a)$ to $\mathcal{A}_0$ and by propagating the concept alongside the first configuration(s) as follows:
$$I \sqcap \overline{C}_{A=2^k-1} \sqsubseteq \bigsqcap_{R_\delta\in\mathcal{R}} \forall R_\delta.I.$$
- The first configuration is modeled by adding the assertion $q_0(a)$ to $\mathcal{A}_0$ and by considering the following GCIs, for all $0 \leq i < k$:
$$\begin{aligned} I \sqcap C_{A=i} &\sqsubseteq \sigma_i, \\ I \sqcap C_{A=k} &\sqsubseteq B, \\ I \sqcap B \sqcap \overline{C}_{A=2^k-1} &\sqsubseteq \bigsqcap_{R_\delta\in\mathcal{R}} \forall R_\delta.B, \end{aligned}$$
where $w = \sigma_0 \dots \sigma_{k-1}$ is the input word.

We finally come to the most involved part, the synchronization of the configurations, which includes the modeling of the transitions.

- To this end, we first introduce the A'-counter, which is increased along the temporal dimension. For every possible value of this counter, there is a time point where a belongs to the concepts from the corresponding subset of $\{A'_0, \dots, A'_{k-1}\}$. This is expressed using the following conjunct of $\phi_{\mathcal{M},w}$:
$$\Box\Box^- \bigwedge_{0\leq i\leq k-1} \Big(\Big(\bigwedge_{0\leq j<i} A'_j(a)\Big) \leftrightarrow \Big(A'_i(a) \leftrightarrow \bigcirc\neg A'_i(a)\Big)\Big)$$
This formula expresses that the i-th bit of the A'-counter is flipped from one world to the next iff all preceding bits are true. Thus, the value of the A'-counter at the next world is equal to the value at the current world incremented by one.

  It is not necessary to initialize this counter to 0 in $\mathcal{A}_0$; we only need to know that all possible counter values are represented in some time point.

- The value of the A'-counter is always shared by all individuals:

$$\Box\Box^-\Big(\bigwedge_{0\leq i\leq k-1} \exists x.A'_i(x) \rightarrow \neg\exists x.\overline{A}'_i(x)\Big).$$

For the application of the A'-counter we now introduce the abbreviation $E_{A,A'}$ describing the equality of the two counters as follows:

$$\begin{aligned} E^i_{A,A'} &\equiv (A_i \sqcap A'_i) \sqcup \left(\overline{A}_i \sqcap \overline{A}'_i\right), \\ E_{A,A'} &\equiv \bigsqcap_{0\leq i<k} E^i_{A,A'}. \end{aligned}$$

Furthermore, we define similar abbreviations as follows:

$$\begin{aligned} E_{A,(A'+1 \bmod 2^k)} \equiv & \bigsqcap_{0\leq j<k} \left(A'_j \sqcap \overline{A}_j\right) \sqcup \\ & \bigsqcup_{0\leq i<k} \left(\bigsqcap_{0\leq j<i} \left(A'_j \sqcap \overline{A}_j\right) \sqcap \overline{A}'_i \sqcap A_i \sqcap \bigsqcap_{i+1\leq j<k} E^j_{A,A'}\right), \\ E_{A,(A'+2 \bmod 2^k)} \equiv & E^0_{A,A'} \sqcap \bigsqcap_{1\leq j<k} \left(A'_j \sqcap \overline{A}_j\right) \sqcup \\ & \bigsqcup_{1\leq i<k} \left(\bigsqcap_{1\leq j<i} \left(A'_j \sqcap \overline{A}_j\right) \sqcap \overline{A}'_i \sqcap A_i \sqcap \bigsqcap_{i+1\leq j<k} E^j_{A,A'}\right). \end{aligned}$$

We now can use the temporal dimension to propagate information from one level of the tree to the next one as outlined above, and hence specify the transitions as follows:

- Symbols not under the head do not change:

$$\begin{aligned} \sigma \sqcap \bigsqcap_{q\in\mathcal{Q}} \overline{q} \sqcap E_{A,A'} &\sqsubseteq \bigsqcap_{R_\delta\in\mathcal{R}} \forall R_\delta.\sigma', \\ \sigma' \sqcap \overline{E}_{A,A'} &\sqsubseteq \bigsqcap_{R_\delta\in\mathcal{R}} \forall R_\delta.\sigma', \\ \sigma' \sqcap E_{A,A'} &\sqsubseteq \sigma. \end{aligned}$$

- We now describe the transitions. Particularly, we explicitly memorize chosen transitions by using the rigid concepts $T_{p,\varrho,M}$—by enforcing them to be satisfied by the elements representing the cells directly right-neighbored to the head position. Indeed, in general, there may be several such cells because we are now at the point where we specify the branching of the tree. Hence, we model the transitions, for all $q \in \mathcal{Q}$ and $\sigma \in \Gamma$, using the following GCIs:

$$q \sqcap \sigma \sqsubseteq \bigsqcup_{\delta \in \Delta(q,\sigma)} \exists R_\delta, \quad \text{if } q \in \mathcal{Q}_\exists$$
$$q \sqcap \sigma \sqsubseteq \bigsqcap_{\delta \in \Delta(q,\sigma)} \exists R_\delta, \quad \text{if } q \in \mathcal{Q}_\forall$$
$$q \sqcap \sigma \sqsubseteq \bigsqcap_{\delta \in \Delta(q,\sigma)} \forall R_\delta.T_\delta$$

- The (possible) replacement of the symbols under the head is now described with the help of the transition concepts $T_{q,\sigma,M}$, for all $q \in \mathcal{Q}$, $\sigma \in \Gamma$, $M \in \{L, R\}$:
$$T_{q,\sigma,M} \sqcap E_{A,(A'+1) \bmod 2^k} \sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.\sigma'.$$
Recall that the transition concepts are only enforced to hold at the cell to the right of the current head position (hence the "+1").

- The state information is similarly propagated as follows, for all $q \in \mathcal{Q}$, $\sigma \in \Gamma$:
$$T_{q,\sigma,R} \sqcap E_{A,A'} \sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.q',$$
$$T_{q,\sigma,L} \sqcap E_{A,(A'+2) \bmod 2^k} \sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.q',$$
$$q' \sqcap \overline{E}_{A,A'} \sqsubseteq \bigsqcap_{R_\delta \in \mathcal{R}} \forall R_\delta.q',$$
$$q' \sqcap E_{A,A'} \sqsubseteq q.$$

We lastly enforce the computation to be an accepting one by disallowing the state q_r entirely using the GCI:
$$q_r \sqsubseteq \bot.$$
Note that this is correct since all the computations of $\mathcal{M}$ are terminating, and each computation must end in a halting configuration (i.e., a configuration with state q_a or q_r). This finishes the definition of the Boolean TCQ $\phi_{\mathcal{M},w}$ and the global ontology $\mathcal{O}_{\mathcal{M},w}$, which consist of the conjuncts and GCIs specified above. We further collect all assertions in the ABox $\mathcal{A}_0$. Given our descriptions above, it is easy to see that the size of $\phi_{\mathcal{M},w}$, $\mathcal{O}_{\mathcal{M},w}$, and $\mathcal{A}_0$ is polynomial in k. Moreover, $\phi_{\mathcal{M},w}$ is satisfiable w.r.t. $\langle \mathcal{O}_{\mathcal{M},w}, (\mathcal{A}_0) \rangle$ iff $\mathcal{M}$ accepts w. □

7.2 Upper Bounds

We now show the corresponding upper bounds.

7.2.1 Regarding Data Complexity

For data complexity, we directly reduce *DL-Lite*$^{\mathcal{H}}_{bool}$ to $\mathcal{ALCH}$, and then reuse the results from [BBL15c].

We first describe the reduction from TCQ entailment in *DL-Lite*$^{\mathcal{H}}_{bool}$ to TCQ entailment in $\mathcal{ALCH}$. It will incur an exponential blowup in the size of the query, but this is irrelevant for results concerning data complexity. Let thus $\mathcal{K} = \langle \mathcal{O}, (\mathcal{A}_i)_{0\leq i\leq n}\rangle$ be a TKB in *DL-Lite*$^{\mathcal{H}}_{bool}$ and ϕ be a TCQ. We construct an $\mathcal{ALCH}$ TKB $\mathcal{K}'$ and a TCQ ϕ' such that $\mathcal{K} \models \phi$ iff $\mathcal{K}' \models \phi'$.

We first extend the set of role names to include all inverse roles R^-, where R or R^- occurs in $\mathcal{O}$. Then, we construct the TKB $\mathcal{K}' := \langle \mathcal{O}', (\mathcal{A}'_i)_{0\leq i\leq n}\rangle$ from $\mathcal{K}$ by replacing all occurrences of concepts of the form $\exists R$, $R \in \mathsf{N}_\mathsf{R}^-$, by $\exists R.\top$, and adding the following axioms:

(i) a GCI $\exists R.(\neg\exists R^-.\top) \sqsubseteq \bot$, for each $R \in \mathsf{N}_\mathsf{R}^-(\mathcal{O})$; and

(ii) an RI $R^- \sqsubseteq S^-$ for each $R \sqsubseteq S \in \mathcal{O}$.

We call a CQ ψ' a *variant* of a CQ ψ if ψ' is obtained from ψ by replacing some role atoms $R(t,t') \in \psi$ by $R^-(t',t)$. We construct ϕ' by replacing every CQ ψ in ϕ by a big disjunction of all variants of ψ. The correctness of this reduction is established next.

Lemma 7.7. *We have $\mathcal{K} \models \phi$ iff $\mathcal{K}' \models \phi'$.*

Proof. ($\Leftarrow$) Let $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ be a model of $\mathcal{K}$ such that $\mathfrak{I} \not\models \phi$. We show that we then have a model $\mathfrak{I}' = (\mathcal{I}'_i)_{i\geq 0}$ of $\mathcal{K}'$ such that $\mathfrak{I}' \not\models \phi'$. Specifically, $\mathfrak{I}'$ has the same domain as $\mathfrak{I}$, interprets all symbols occurring in $\mathcal{K}$ as $\mathfrak{I}$ does, and, for all $\mathcal{I}'_i$, the interpretation of role names R^- such that $R^- \in \mathsf{N}_\mathsf{R}(\mathcal{O}') \setminus \mathsf{N}_\mathsf{R}(\mathcal{O})$, is equal to $(R^-)^{\mathcal{I}_i}$.

Given this definition of $\mathfrak{I}'$, we obviously have that $\mathcal{I}'_i \models \mathcal{A}'_i$, $0 \leq i \leq n$, since $\mathcal{I}_i \models \mathcal{A}_i$. The same holds for the GCIs and RIs that are contained in $\mathcal{O}$. Moreover, it is easy to see that the new GCIs and RIs are satisfied, too. We thus have $\mathfrak{I}' \models \mathcal{K}'$.

We now assume that $\mathfrak{I}' \models \phi'$, by contradiction. Given the construction of ϕ', we first show that, for every CQ ψ in ϕ which is replaced by a disjunction α in ϕ', $\mathcal{I}'_i \models \alpha$ leads to $\mathcal{I}_i \models \psi$, for all $i \geq 0$. Let thus π be a homomorphism of some CQ ψ' in an arbitrary such disjunction α into some $\mathcal{I}'_i$, and ψ be the CQ that was replaced by α. ψ and ψ' thus only differ in the role atoms. Let $R(t,t')$ be an atom in ψ and $S^-(t',t)$ be the corresponding replacement in ψ' (i.e., we have $\mathcal{O} \models S \sqsubseteq R$). By (ii) and $\mathfrak{I}' \models \mathcal{O}'$, we obtain $(\pi(t'),\pi(t)) \in (R^-)^{\mathcal{I}'_i}$ from $(\pi(t'),\pi(t)) \in (S^-)^{\mathcal{I}'_i}$. Hence, we also have $(\pi(t),\pi(t')) \in R^{\mathcal{I}_i}$, by construction, which yields that π is also a homomorphism of ψ into $\mathcal{I}_i$. Considering the other direction, we trivially have that $\mathcal{I}'_i \not\models \alpha$ leads to $\mathcal{I}_i \not\models \psi$, for all $i \geq 0$, given our construction. By induction on the shape of ϕ, it now can be easily shown that $\mathfrak{I} \models \phi$ follows, which contradicts the assumption.

($\Rightarrow$) Let now $\mathfrak{I}' = (\mathcal{I}'_i)_{i\geq 0}$ be a model of $\mathcal{K}'$ such that $\mathfrak{I}' \not\models \phi'$. We show this direction similarly by constructing a model $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ of $\mathcal{K}$ such that $\mathfrak{I} \not\models \phi$. In particular, we assume $\mathfrak{I}$ to have the same domain as $\mathfrak{I}'$, to interpret all concept names as $\mathfrak{I}'$ does, and to interpret all role names $R \in \mathsf{N_R}(\mathcal{O})$ such that $R^{\mathcal{I}_i} = R^{\mathcal{I}'_i} \cup \{(e, e') \mid (e', e) \in (R^-)^{\mathcal{I}'_i}\}$, for all $i \geq 0$.

The latter definition particularly yields that $(e, e') \in R^{\mathcal{I}_i}$ if $(e, e') \in R^{\mathcal{I}'_i}$, and $(e, e') \in (R^-)^{\mathcal{I}_i}$ if $(e, e') \in (R^-)^{\mathcal{I}'_i}$, $i \geq 0$. Together with (i) and the definition of $R^{\mathcal{I}_i}$, we thus obtain that $e \in (\exists R)^{\mathcal{I}_i}$ iff $e \in (\exists R)^{\mathcal{I}'_i}$, for $R \in \mathsf{N}^-_\mathsf{R}(\mathcal{O})$. Regarding the ABoxes, we thus must specially consider negative role assertions and assume to have $\neg R^-(b, a) \in \mathcal{A}_i$, $(b, a) \notin (R^-)^{\mathcal{I}'_i}$, but $(a, b) \in R^{\mathcal{I}_i}$. The latter implies that $(a, b) \in R^{\mathcal{I}'_i}$, by the definition of $R^{\mathcal{I}_i}$. However, this directly yields a contradiction since we assumed to have $\neg R(a, b) \in \mathcal{A}_i$ if $\neg R^-(b, a) \in \mathcal{A}_i$. Then, it is easy to see that we get $\mathcal{I}_i \models \mathcal{A}_i$, $0 \leq i \leq n$, since $\mathcal{I}'_i \models \mathcal{A}_i$. The above observation about concepts of the form $e \in (\exists R)^{\mathcal{I}_i}$, $R \in \mathsf{N}^-_\mathsf{R}(\mathcal{O})$, and the fact that the interpretations $\mathcal{I}'_i$ satisfy $\mathcal{O}$ further yields $\mathcal{I}_i \models \mathcal{O}$. Lastly, we consider an arbitrary RI $R \sqsubseteq S$ in $\mathcal{O}$ and $(e, e') \in R^{\mathcal{I}_i}$. If $(e, e') \in R^{\mathcal{I}'_i}$, then $\mathcal{I}'_i \models \mathcal{O}$ implies $(e, e') \in S^{\mathcal{I}'_i} \subseteq S^{\mathcal{I}_i}$. Otherwise, we must have $(e', e) \in (R^-)^{\mathcal{I}'_i}$ and, by (ii) (i.e., we have $R^- \sqsubseteq S^- \in \mathcal{O}'$) and $\mathcal{I}'_i \models \mathcal{O}'$ get $(e', e) \in (S^-)^{\mathcal{I}'_i}$. But then, we have $(e, e') \in S^{\mathcal{I}_i}$, as well. Hence, we have $\mathfrak{I} \models \mathcal{K}$.

As above, we now assume that $\mathcal{I} \models \phi$, by contradiction. Given the construction of ϕ', we regard an arbitrary CQ ψ in ϕ replaced by a TCQ α in ϕ' (i.e., α is a disjunction of CQs). If $\mathcal{I}' \models \alpha$, then there must be a homomorphism from a disjunct ψ' of α into $\mathcal{I}'$, and thus obviously $\mathcal{I} \models \psi$ by the constructions of α and $\mathcal{I}$. Let now π be a homomorphism of ψ into $\mathcal{I}$. Then, for every role atom $R(t, t')$ in ψ, we must either have $(\pi(t), \pi(t')) \in R^{\mathcal{I}'}$ or $(\pi(t'), \pi(t)) \in (R^-)^{\mathcal{I}'}$, by the construction of $\mathcal{I}$. But then, π is also a homomorphism of the variant ψ' contained in α, which contains the corresponding combination of role atoms, into $\mathcal{I}'$. This yields $\mathcal{I}' \models \alpha$. By induction, it now can be easily shown that $\mathcal{I}' \models \phi'$ follows, which contradicts the assumption. □

This reduction allows us to use the results for TCQ entailment in $\mathcal{SHQ}$ [BBL15c] to show the following upper bounds.

Theorem 7.8. *TCQ entailment in DL-Lite$^{\mathcal{H}}_{bool}$ w.r.t. data complexity is*

- *in* co-NP *if* $\mathsf{N_{RR}} = \emptyset$ *and*
- *in* ExpTime *if* $\mathsf{N_{RR}} \neq \emptyset$.

Note that for $\mathcal{ALCH}$ a tight upper bound for the case $\mathsf{N_{RR}} \neq \emptyset$ w.r.t. data complexity is still open. Hence, we here also have a gap between co-NP and ExpTime.

Unfortunately, this reduction is not directly applicable for combined complexity because ϕ' is exponentially larger than ϕ.

7.2.2 Regarding Combined Complexity

Regarding combined complexity, the 2-ExpTime upper bound for $DL\text{-}Lite^{\mathcal{H}}_{bool}$ and the case that $\mathsf{N_{RR}} \neq \emptyset$ follows from the case of $\mathcal{SHIQ}$ [BBL15a, BBL15b].

To show the remaining results for $DL\text{-}Lite_{bool}$, recall that we only need to consider $DL\text{-}Lite_{krom}$ (see Corollary 7.3). We use the techniques applied in [BBL15c] to solve TCQ entailment w.r.t. TKBs formulated in $\mathcal{SHQ}$. The latter reduce the problem to one or several atemporal satisfiability problems for conjunctions of CQ-literals, similar to those in Definition 4.2. Hence, we first need to establish the complexity of this problem for $DL\text{-}Lite_{krom}$.

Lemma 7.9. *Let ϕ be a Boolean conjunction of CQ-literals and $\mathcal{K} = \langle \mathcal{O}, \mathcal{A} \rangle$ be an (atemporal) KB in $DL\text{-}Lite_{krom}$. Then the satisfiability of ϕ w.r.t. $\mathcal{K}$ can be decided in* ExpTime *w.r.t. combined complexity.*

Proof. Following the proof of [BBL15c, Theorem 4.1.], we reduce this problem to exponentially many UCQ non-entailment problems.

Let $\phi = \rho_1 \wedge \cdots \wedge \rho_\ell \wedge \neg\sigma_1 \wedge \cdots \wedge \neg\sigma_m$, where $\rho_1, \ldots, \rho_\ell, \sigma_1, \ldots, \sigma_m$ are Boolean CQs. We instantiate the variables in $\rho_1, \ldots, \rho_\ell$ with new individual names and collect the resulting assertions into the ABox $\mathcal{A}'$. Then there is a model $\mathcal{I}$ of $\mathcal{K}$ which satisfies ϕ iff there is a model $\mathcal{I}'$ of $\langle \mathcal{O}, \mathcal{A} \cup \mathcal{A}' \rangle$ which satisfies $\neg\sigma_1 \wedge \cdots \wedge \neg\sigma_m$ (to satisfy the UNA w.r.t. the new individuals, we can copy the affected domain elements). The latter is the non-entailment problem $\langle \mathcal{O}, \mathcal{A} \cup \mathcal{A}' \rangle \not\models \sigma_1 \vee \cdots \vee \sigma_m$. Since UCQ non-entailment for frontier-one disjunctive inclusion dependencies (DIDs), and hence in $DL\text{-}Lite_{krom}$, can be decided in ExpTime according to [BMP13, Theorem 8], we obtain the result stated above. □

Given Lemma 7.9 together with Corollary 7.3, we now can prove the following.

Theorem 7.10. *For combined complexity, TCQ entailment in $DL\text{-}Lite_{bool}$ is*

- *in* ExpTime *if* $\mathsf{N_{RC}} = \emptyset$ *and* $\mathsf{N_{RR}} = \emptyset$*, and*
- *in* co-NExpTime *if* $\mathsf{N_{RR}} = \emptyset$*.*

Proof. By Corollary 7.3, it suffices to show the upper bounds for $DL\text{-}Lite_{krom}$. In line with [BBL15c], we follow the basic approach of Lemma 4.3, considering the satisfiability problem of a TCQ ϕ w.r.t. a TKB $\mathcal{K}$, and begin with the case that $\mathsf{N_{RC}} = \emptyset$ and $\mathsf{N_{RR}} = \emptyset$.

- All possible mappings ι can be enumerated in exponential time.
- Moreover, the test if ι is compatible with an LTL-model of ϕ^{p} can be done in ExpTime, according to [BBL15c, Lemma 4.12].
- To check r-satisfiability of a set $\mathcal{S} = \{X_1, \dots, X_k\} \subseteq 2^{\{p_1,\dots,p_m\}}$ w.r.t. a ι and $\mathcal{O}$, it clearly suffices to check satisfiability of the (possibly exponentially many) conjunctions χ_i, $1 \leq i \leq k$, and $\chi_{\iota(i)} \wedge \bigwedge_{\alpha\in\mathcal{A}_i} \alpha$, $0 \leq i \leq n$, w.r.t. $\mathcal{O}$ individually.[15] This is because, without rigid names, it is impossible to enforce any dependency between the sets $X \in \mathcal{S}$. Hence, it suffices to define $\mathcal{S}$ as the set of *all* sets X_i for which χ_i is satisfiable w.r.t. $\mathcal{O}$. According to Lemma 7.9, the latter can be decided in ExpTime.

It can be easily seen that the consideration of all possible ι and our deterministic definition of the maximal possible $\mathcal{S}$ suffices to satisfy Lemma 4.3. We thus can decide TCQ satisfiability (and entailment) in ExpTime.

We now consider the case that only $\mathsf{N_{RR}} = \emptyset$, but some concept names may be rigid. As in [BBL15c], we can assume w.l.o.g. that the input contains exactly one empty ABox, as under combined complexity the ABoxes can be encoded into the TCQ. Hence, we can disregard the mapping ι in the following. We can obviously guess $\mathcal{S}$ and check t-satisfiability in exponential time. For the r-satisfiability test, we additionally need to guess a set $\mathcal{D} \subseteq \mathsf{N_{RR}}(\mathcal{O})$ that specifies the combinations of rigid names that are allowed to occur in a model, and a mapping $\tau\colon \mathsf{N_I}(\phi) \to \mathcal{D}$ that fixes the behavior of the rigid names on the named domain elements. We define the polynomial-sized ontology

$$\mathcal{O}_\tau := \{A_{\tau(a)} \equiv C_{\tau(a)} \mid a \in \mathsf{N_I}(\phi)\}$$

and the additional conjunction

$$\rho_\tau := \bigwedge_{a\in\mathsf{N_I}(\phi)} A_{\tau(a)}(a),$$

where C_Y, $Y \subseteq \mathsf{N_{RR}}(\mathcal{O})$, is defined as

$$C_Y := \bigsqcap_{A\in Y} A \sqcap \bigsqcap_{A\in\mathsf{N_{RR}}(\mathcal{O})\setminus Y} \overline{A}.$$

We further say that an interpretation $\mathcal{I}$ *respects* $\mathcal{D}$ if

$$\mathcal{D} = \{Y \subseteq \mathsf{N_{RR}}(\mathcal{O}) \mid \text{there is a } d \in \Delta^{\mathcal{I}} \text{ with } d \in (C_Y)^{\mathcal{I}}\}.$$

[15] We can assume that all of these models have the same domain since their domains can be assumed to be countably infinite by the Löwenheim-Skolem theorem, and that all individual names are interpreted by the same domain elements in all models.

It is shown in [BBL15c, Lemma 6.2] that $\mathcal{S}$ is r-satisfiable w.r.t. $\mathcal{K}$ iff there are $\mathcal{D}$ and τ as above such that each conjunction $\chi_i \wedge \rho_\tau$ has a model w.r.t. $\mathcal{O} \cup \mathcal{O}_\tau$ that respects $\mathcal{D}$. The proof of this result is given for $\mathcal{SHQ}$, but it remains the same for $DL\text{-}Lite_{krom}$. If not for the condition involving $\mathcal{D}$, the claimed result would now follow from Lemma 7.9. For $\mathcal{D}$, we have to consider the proof of [BMP13, Theorem 8] in more detail. There, an exponentially large looping tree automaton is constructed that recognizes exactly those (forest-shaped) canonical models of the KB that do not satisfy the given UCQ. This automaton is easily modified to accept arbitrary models (the restriction to tree-shaped models is without loss of generality even for UCQ entailment). We can further restrict the state set to consider only models where every domain element satisfies some C_Y with $Y \in \mathcal{D}$. To ensure that each $Y \in \mathcal{D}$ is represented somewhere in the model, we additionally check emptiness for the variants of this automaton using the ABoxes $\{A(a) \mid A \in Y\} \cup \{\overline{A}(a) \mid A \in \mathsf{N}_{\mathsf{RR}}(\mathcal{O}) \setminus Y\}$, where a is a fresh individual name. The disjoint union of all resulting interpretations will still be a model of the original KB that does not satisfy the UCQ. The complexity result follows from the fact that emptiness of looping tree automata can be decided in polynomial time [VW86]. □

8 Conclusions

We have analyzed the computational complexity of TCQ entailment in several members of the *DL-Lite* family of Description Logics. As it can be seen in Table 1.1, many of these fragments turned out to be very complex. Nevertheless, for several others, we obtained encouraging results, which are even better than those for $\mathcal{EL}$. Especially the data complexity of ALogTime implies that it might be possible to solve the entailment problem by applying the combined approach of [KLT+10] or by rewriting the TCQ into a Datalog query to be evaluated over a database [DEGV97].

We further showed that the combined complexity of PSpace inherited from LTL does not increase—even if rigid role names are considered. If we make the reasonable assumption that all relevant information about these names (e.g., which patients have no allergy and thus do not belong to the rigid concept $\exists\mathsf{AllergyTo}$) is available before query answering, then we do not need to guess the ABox type $\mathcal{A}_{\mathsf{R}}$. It remains to be seen whether existing PSpace-algorithms for LTL [GO01] can be efficiently combined with reasoning procedures for *DL-Lite* [KLT+10].

In future work, it would be worth to study other variants of *DL-Lite* [ACKZ09] since it might be possible to go beyond $DL\text{-}Lite^{\mathcal{H}}_{horn}$ while keeping its complexity. We could also combine our approach with other temporal query formalisms based on *DL-Lite* [AKL+07, AKK+14, AKK+15], and investigate how to transfer and combine existing constructions and results. On the practical side, it would be interesting to see how TCQs perform in applications; some prototype implementations have already been described [THÖ15].

References

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach.* Cambridge University Press, 2009.

[ABM+14] Alessandro Artale, Davide Bresolin, Angelo Montari, Guido Sciavicco, and Vladislav Ryzhikov. *DL-Lite* and interval temporal logics: A marriage proposal. In Torsten Schaub, editor, *Proc. of the 21st Eur. Conf. on Artificial Intelligence (ECAI'14)*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 957–958. IOS Press, 2014.

[ACKZ09] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyaschev. The *DL-Lite* family and relations. *Journal of Artificial Intelligence Research*, 36:1–69, 2009.

[AHV95] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases.* Addison-Wesley, 1995.

[AKK+14] Alessandro Artale, Roman Kontchakov, Alisa Kovtunova, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev. Temporal OBDA with LTL and *DL-Lite.* In Meghyn Bienvenu, Magdalena Ortiz, Riccardo Rosati, and Mantas Šimkus, editors, *Proc. of the 27th Int. Workshop on Description Logics (DL'14)*, volume 1193 of *CEUR Workshop Proceedings*, pages 21–32, 2014.

[AKK+15] Alessandro Artale, Roman Kontchakov, Alisa Kovtunova, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev. First-order rewritability of ontology-mediated temporal queries. In Qiang Yang, editor, *Proc. of the 24th Int. Joint Conf. on Artificial Intelligence (IJCAI'15)*, pages 2706–2712. AAAI Press, 2015.

[AKL+07] Alessandro Artale, Roman Kontchakov, Carsten Lutz, Frank Wolter, and Michael Zakharyaschev. Temporalising tractable description logics. In Valentin Goranko and X. Sean Wang, editors, *Proc. of the 14th Int. Symp. on Temporal Representation and Reasoning (TIME'07)*, pages 11–22. IEEE Press, 2007.

[AKRZ14] Alessandro Artale, Roman Kontchakov, Vladislav Ryzhikov, and Michael Zakharyaschev. A cookbook for temporal conceptual data modelling with description logics. *ACM Transactions on Computational Logic*, 15(3):25, 2014.

[BAC10] Elena Botoeva, Alessandro Artale, and Diego Calvanese. Query rewriting in *DL-Lite*$_{horn}^{(\mathcal{HN})}$. In Volker Haarslev, David Toman, and Grant Weddell, editors, *Proc. of the 23rd Int. Workshop on Description Logics (DL'10)*, volume 573 of *CEUR Workshop Proceedings*, pages 267–278, 2010.

[BBL13] Franz Baader, Stefan Borgwardt, and Marcel Lippmann. Temporalizing ontology-based data access. In Maria Paola Bonacina, editor, *Proc. of the 24th Int. Conf. on Automated Deduction (CADE'13)*, volume 7898 of *Lecture Notes in Computer Science*, pages 330–344. Springer-Verlag, 2013.

[BBL15a] Franz Baader, Stefan Borgwardt, and Marcel Lippmann. Temporal conjunctive queries in expressive description logics with transitive roles. In Bernhard Pfahringer and Jochen Renz, editors, *Proc. of the 28th Australasian Joint Conf. on Artificial Intelligence (AI'15)*, volume 9457 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 2015. To appear.

[BBL15b] Franz Baader, Stefan Borgwardt, and Marcel Lippmann. Temporal conjunctive queries in expressive DLs with non-simple roles. LTCS-Report 15-17, Chair for Automata Theory, TU Dresden, Germany, 2015. See `https://ddll.inf.tu-dresden.de/web/Ver%C3%B6ffentlichungen/2015/en#Technical_Reports`.

[BBL15c] Franz Baader, Stefan Borgwardt, and Marcel Lippmann. Temporal query entailment in the description logic $\mathcal{SHQ}$. *Journal of Web Semantics*, 33:71–93, 2015.

[BCM+03] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[BCST92] David A. Mix Barrington, Kevin Compton, Howard Straubing, and Denis Thérien. Regular languages in NC^1. *Journal of Computer and System Sciences*, 44(3):478–499, 1992.

[BGL12] Franz Baader, Silvio Ghilardi, and Carsten Lutz. LTL over description logic axioms. *ACM Transactions on Computational Logic*, 13(3):21:1–21:32, 2012.

[BLT15] Stefan Borgwardt, Marcel Lippmann, and Veronika Thost. Temporalizing rewritable query languages over knowledge bases. *Journal of Web Semantics*, 33:50–70, 2015.

[BMP13] Pierre Bourhis, Michael Morak, and Andreas Pieris. The impact of disjunction on query answering under guarded-based existential rules. In Francesca Rossi, editor, *Proc. of the 23rd Int. Joint Conf. on Artificial Intelligence (IJCAI'13)*, pages 796–802. AAAI Press, 2013.

[BMP14] Pierre Bourhis, Michael Morak, and Andreas Pieris. Acyclic query answering under guarded disjunctive existential rules and consequences to DLs. In Meghyn Bienvenu, Magdalena Ortiz, Riccardo Rosati, and Mantas Šimkus, editors, *Proc. of the 27th Int. Workshop on Description Logics (DL'14)*, volume 1193 of *CEUR Workshop Proceedings*, pages 100–111, 2014.

[BT15a] Stefan Borgwardt and Veronika Thost. LTL over $\mathcal{EL}$ axioms. LTCS-Report 15-07, Chair for Automata Theory, TU Dresden, Germany, 2015. See `https://ddll.inf.tu-dresden.de/web/Ver%C3%B6ffentlichungen/2015/en#Technical_Reports`.

[BT15b] Stefan Borgwardt and Veronika Thost. Temporal query answering in $\mathcal{EL}$. LTCS-Report 15-08, Chair for Automata Theory, TU Dresden, Germany, 2015. See `https://ddll.inf.tu-dresden.de/web/Ver%C3%B6ffentlichungen/2015/en#Technical_Reports`.

[BT15c] Stefan Borgwardt and Veronika Thost. Temporal query answering in the description logic $\mathcal{EL}$. In Qiang Yang, editor, *Proc. of the 24th Int. Joint Conf. on Artificial Intelligence (IJCAI'15)*, pages 2819–2825. AAAI Press, 2015.

[CDGL+05] Diego Calvanese, Giuseppe De Giacomo, Domenico Lemho, Maurizio Lenzerini, and Riccardo Rosati. *DL-Lite*: Tractable description logics for ontologies. In Manuela M. Veloso and Subbaro Kambhampati, editors, *Proc. of the 20th Nat. Conf. on Artificial Intelligence (AAAI'05)*, pages 602–607. AAAI Press, 2005.

[CDGL+09] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Mariano Rodriguez-Muro, and Riccardo Rosati. Ontologies and databases: The *DL-Lite* approach. In Sergio Tessaris, Enrico Franconi, Thomas Eiter, Claudio Gutierrez, Siegfried Handschuh, Marie-Christine Rousset, and Renate A. Schmidt, editors, *Reasoning Web. Semantic Technologies for Information Systems*, volume 5689 of *Lecture Notes in Computer Science*, pages 255–356. Springer Berlin Heidelberg, 2009.

[CDL+07] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *Journal of Automated Reasoning*, 39(3):385–429, 2007.

[CKS81] Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. Alternation. *Journal of the ACM*, 28(1):114–133, 1981.

[CM77] Ashok K. Chandra and Philip M. Merlin. Optimal implementation of conjunctive queries in relational data bases. In *Proc. of the 9th Annual ACM Symp. on Theory of Computing (STOC'77)*, pages 77–90. ACM, 1977.

[DEGV97] Evgeny Dantsin, Thomas Eiter, Georg Gottlob, and Andrei Voronkov. Complexity and expressive power of logic programming. In *Proc. of the 12th Annual IEEE Conf. on Computation Complexity (CCC'97)*, pages 82–101. IEEE Press, 1997.

[Gab87] Dov M. Gabbay. The declarative past and imperative future: Executable temporal logic for interactive systems. In *Temporal Logic in Specification*, pages 409–448. Springer-Verlag, 1987.

[GBJS15] Víctor Gutiérrez-Basulto, Jean Christoph Jung, and Thomas Schneider. Lightweight temporal description logics with rigid roles and restricted TBoxes. In Qiang Yang, editor, *Proc. of the 24th Int. Joint Conf. on Artificial Intelligence (IJCAI'15)*, pages 3015–3021. AAAI Press, 2015.

[GHJR+15] Martin Giese, Peter Haase, Ernesto Jiménez-Ruiz, Davide Lanti, Özgür Özçep, Martin Rezk, Riccardo Rosati, Ahmet Soylu, Guillermo Vega-Gorgojo, Arild Waaler, and Guohui Xiao. Optique: Zooming in on Big Data. *IEEE Computer*, 48(3):60–67, 2015.

[GJS14] Víctor Gutiérrez-Basulto, Jean Christoph Jung, and Thomas Schneider. Lightweight description logics and branching time: A troublesome marriage. In Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, editors, *Proc. of the 14th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'14)*. AAAI Press, 2014.

[GO01] Paul Gastin and Denis Oddoux. Fast LTL to Büchi automata translation. In Gérard Berry, Hubert Comon, and Alain Finkel, editors, *Proc. of the 13th Int. Conf. on Computer Aided Verification (CAV'01)*, volume 2102 of *Lecture Notes in Computer Science*, pages 53–65. Springer-Verlag, 2001.

[KLT+10] Roman Kontchakov, Carsten Lutz, David Toman, Frank Wolter, and Michael Zakharyaschev. The combined approach to query answering in *DL-Lite*. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczynski, editors, *Proc. of the 12th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'10)*, pages 247–257. AAAI Press, 2010.

[KRH13] Markus Krötzsch, Sebastian Rudolph, and Pascal Hitzler. Complexities of Horn description logics. *ACM Transactions on Computational Logic*, 14(1):2:1–2:36, 2013.

[LWZ08] Carsten Lutz, Frank Wolter, and Michael Zakharyaschev. Temporal description logics: A survey. In Stéphane Demri and Christian S. Jensen, editors, *Proc. of the 15th Int. Symp. on Temporal Representation and Reasoning (TIME'08)*, pages 3–14. IEEE Press, 2008.

[MBIS90] David A. Mix Barrington, Neil Immerman, and Howard Straubing. On uniformity within NC^1. *Journal of Computer and System Sciences*, 41(3):274–306, 1990.

[PLC+08] Antonella Poggi, Domenico Lembo, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Linking data to ontologies. *Journal on Data Semantics*, X:133–173, 2008.

[Pnu77] Amir Pnueli. The temporal logic of programs. In *Proc. of the 18th Annual Symp. on Foundations of Computer Science (SFCS'77)*, pages 46–57. IEEE Press, 1977.

[RG10] Sebastian Rudolph and Birte Glimm. Nominals, inverses, counting, and conjunctive queries or: Why infinity is your friend! *Journal of Artificial Intelligence Research*, 39(1):429–481, 2010.

[Sav70] Walter J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *Journal of Computer and System Sciences*, 4(2):177–192, 1970.

[SC85] A. Prasad Sistla and Edmund M. Clarke. The complexity of propositional linear temporal logics. *Journal of the ACM*, 32(3):733–749, 1985.

[Tes01] Sergio Tessaris. *Questions and Answers: Reasoning and Querying in Description Logic.* PhD thesis, University of Manchester, UK, 2001.

[THÖ15] Veronika Thost, Jan Holste, and Özgür Özçep. On implementing temporal query answering in *DL-Lite* (extended abstract). In Diego Calvanese and Boris Konev, editors, *Proc. of the 28th Int. Workshop on Description Logics (DL'15)*, volume 1350 of *CEUR Workshop Proceedings*, pages 552–555, 2015.

[VW86] Moshe Y. Vardi and Pierre Wolper. Automata-theoretic techniques for modal logics of programs. *Journal of Computer and System Sciences*, 32(2):183–221, 1986.