Tractable Queries for Lightweight Description Logics

Meghyn Bienvenu
Laboratoire de Recherche en Informatique
CNRS & Université Paris Sud, France

Magdalena Ortiz
Mantas Šimkus
Guohui Xiao
Institute of Information Systems
Vienna University of Technology, Austria

Abstract

It is a classic result in database theory that conjunctive query (CQ) answering, which is NP-complete in general, is feasible in polynomial time when restricted to acyclic queries. Subsequent results identified more general structural properties of CQs (like bounded treewidth) which ensure tractable query evaluation. In this paper, we lift these tractability results to knowledge bases formulated in the lightweight description logics DL-Lite and $\mathcal{ELH}$. The proof exploits known properties of query matches in these logics and involves a query-dependent modification of the data. To obtain a more practical approach, we propose a concrete polynomial-time algorithm for answering acyclic CQs based on rewriting queries into datalog programs. A preliminary evaluation suggests the interest of our approach for handling large acyclic CQs.

1 Introduction

Conjunctive queries (CQs) form a natural and important class of relational databases queries. In the description logic (DL) research community, there has been increasing interest in the problem of retrieving the answers to a CQ while taking into account the knowledge specified by a DL ontology [Calvanese *et al.*, 2007; Lutz *et al.*, 2009]. The use of an ontology typically leads to an increase in the complexity of CQ answering[1] compared to the relational database setting. Indeed, for the so-called *expressive DLs*, CQ answering is co-NP hard in data complexity (that is, when the ontology and query are considered fixed, and the complexity is measured in the size of the data only), in contrast to the AC_0 upper bound for standard databases. In combined complexity (that is, when the complexity is measured in terms of the combined sizes of the query, ontology, and data), rather than NP-complete, the problem is at least ExpTime-hard, and often requires double-exponential time (cf. the survey [Ortiz and Simkus, 2012]). Since such high complexity is prohibitive for data-rich applications, most work in the area focuses on ontologies formulated in *lightweight DLs* of the DL-Lite [Calvanese *et al.*, 2007] and $\mathcal{EL}$ [Baader *et al.*, 2005] families, which enjoy better computational properties. Indeed, CQ answering for DL-Lite knowledge bases has the same data and combined complexity as for plain databases, whereas for $\mathcal{EL}$, CQ answering is P-complete in data complexity, but remains NP-complete in combined complexity.

A classic result in database theory states that CQ answering becomes feasible in polynomial time when restricted to the class of *acyclic CQs* [Yannakakis, 1981]. Later investigations lead to the identification of more general structural properties, such as bounded treewidth, query width, or hypertree width [Chekuri and Rajaraman, 1997; Gottlob *et al.*, 1999], which guarantee tractable CQ answering. Since the NP-hardness in combined complexity of CQ answering in DLs is a direct consequence of the analogous result for relational databases, it seems natural to ask whether these tractability results also transfer to the DL setting. This would be very desirable since it is likely that most of the queries that will actually occur in applications are acyclic. While there are no collections of real-world CQs that can be used to support this claim in the DL setting, one can find some compelling evidence by looking at the closely related setting of SPARQL queries over RDF data, where it has been reported that acyclic queries (in fact, acyclic conjunctive graph patterns) comprise more than 99% of the queries in a log of around three million queries posed to the DBpedia endpoint [Picalausa and Vansummeren, 2011]. Unfortunately, lifting positive results from databases to the DL setting is often not possible, even when one considers lightweight DLs. For instance, for the logic DL-Lite$_\mathcal{R}$, which underlies the QL profile of the OWL 2 standard [OWL Working Group, 2009], CQ answering was recently shown to be NP-hard already for acyclic queries [Kikot *et al.*, 2011].

In this paper, we show that for plain DL-Lite (without role hierarchies) and for $\mathcal{EL}$, the picture is brighter. Specifically, all polynomial-time upper bounds for classes of CQs known from relational databases carry over to DL-Lite. In the case of the $\mathcal{EL}$ family, we get polynomiality even for $\mathcal{ELH}$, thus showing that role hierarchies alone are not the culprit for the loss of tractability. Although this general tractability result relies on known properties of the logics, to our knowledge, it has not been pointed out before. The proof involves a polynomial reduction from the problem of answering a given CQ over a knowledge base $\mathcal{K}$ to answering the same CQ over a database that results from a polynomial expansion of the

[1] When talking about the complexity of CQ answering, we mean the complexity of the *query output tuple problem*, that is, to decide whether a given tuple is in the answer of a CQ.

dataset in $\mathcal{K}$. The algorithm arising from this reduction has a disadvantage: it involves a query-dependent expansion of the data, which may be undesirable in many settings. Hence, we also propose an alternative polynomial-time algorithm for acyclic CQs, based on a rewriting into datalog. We have implemented a simple prototype of the approach, and it shows promising results for answering large acyclic CQs.

2 Preliminaries

Description Logics We briefly recall the syntax and semantics of DL-Lite$_\mathcal{R}$ [Calvanese *et al.*, 2007], $\mathcal{ELH}$ [Baader *et al.*, 2005], and their sublogics DL-Lite and $\mathcal{EL}$. Let $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_I}$ be countably infinite sets of concept names, role names, and individuals, respectively, and let $\overline{\mathsf{N_R}} = \mathsf{N_R} \cup \{r^- \mid r \in \mathsf{N_R}\}$ be the set of *(complex) roles*. For $R \in \overline{\mathsf{N_R}}$, R^- denotes r^- if $R = r \in \mathsf{N_R}$, and r if $R = r^-$.

An *ABox* is a finite set of *assertions* of the forms $A(b)$ and $r(b, c)$ with $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $b, c \in \mathsf{N_I}$. A *TBox* is a finite set of *axioms*, whose form depends on the particular DL. In DL-Lite, TBox axioms are *concept inclusions* of the form $C_1 \sqsubseteq C_2$, with $C_1 = B_1$ and $C_2 = (\neg)B_2$ for B_1, B_2 of the form $A \in \mathsf{N_C}$ or $\exists R$ with $R \in \overline{\mathsf{N_R}}$. DL-Lite$_\mathcal{R}$ TBoxes may also contain *role inclusions* of the form $R_1 \sqsubseteq (\neg)R_2$ with $R_1, R_2 \in \overline{\mathsf{N_R}}$. In $\mathcal{EL}$, TBoxes consist of *concept inclusions* $C_1 \sqsubseteq C_2$, but in this case C_1, C_2 may be *complex concepts* constructed according to the syntax $C := \top \mid A \mid C \sqcap C \mid \exists r.C$. $\mathcal{ELH}$ TBoxes additionally allow role inclusions of the form $r_1 \sqsubseteq r_2$, similarly to DL-Lite$_\mathcal{R}$, but with $r_1, r_2 \in \mathsf{N_R}$. A DL *knowledge base (KB)* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and ABox $\mathcal{A}$.

The semantics of DL KBs is defined in terms of *(DL) interpretations* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where the *domain* $\Delta^\mathcal{I}$ is a non-empty set, and the *interpretation function* $\cdot^\mathcal{I}$ maps each $a \in \mathsf{N_I}$ to an object $a^\mathcal{I} \in \Delta^\mathcal{I}$, each $A \in \mathsf{N_C}$ to a set $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$, and each $r \in \mathsf{N_R}$ to a binary relation $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. The function $\cdot^\mathcal{I}$ is extended to complex concepts and roles in the usual way, see [Calvanese *et al.*, 2007; Baader *et al.*, 2005] for details. We say that $\mathcal{I}$ satisfies an axiom $P_1 \sqsubseteq P_2$ if $P_1^\mathcal{I} \subseteq P_2^\mathcal{I}$), and it satisfies an assertion $A(a)$ (resp. $r(a, b)$) if $a^\mathcal{I} \in A^\mathcal{I}$ (resp. $(a^\mathcal{I}, b^\mathcal{I}) \in r^\mathcal{I}$). Finally, $\mathcal{I}$ is a *model* of a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ if it satisfies every axiom in $\mathcal{T}$ and assertion in $\mathcal{A}$. $\mathcal{K}$ is *consistent* if it admits some model. We use $\mathsf{Ind}(\mathcal{A})$ for the individuals occurring in $\mathcal{A}$. We let $\mathcal{I}_\mathcal{A}$ be the interpretation with $\Delta^\mathcal{I} = \mathsf{Ind}(\mathcal{A})$ and such that (i) $c \in A^\mathcal{I}$ iff $A(c) \in \mathcal{A}$, and (ii) $(c, d) \in r^\mathcal{I}$ iff $r(c, d) \in \mathcal{A}$.

Queries We recall *non-recursive datalog queries* and the more restricted *conjunctive queries*, cf. [Levy and Rousset, 1998]. Let $\mathsf{N_V}$ and $\mathsf{N_D}$ be countably infinite sets of *variables* and *datalog relations*, respectively. Each $\sigma \in \mathsf{N_D}$ has an associated non-negative integer *arity*. *Atoms* are expressions of the form $p(\vec{x})$, where $\vec{x} \in (\mathsf{N_V})^n$, and (i) $p \in \mathsf{N_C}$ and $n = 1$, (ii) $p \in \overline{\mathsf{N_R}}$ and $n = 2$, or (iii) $p \in \mathsf{N_D}$ and n is the arity of p. Atoms of the form (i) and (ii) are called *DL-atoms*.

A *rule* ρ is an expression of the form $h(\vec{x}) \leftarrow \alpha_1, \ldots, \alpha_m$, where $h(\vec{x}), \alpha_1, \ldots, \alpha_m$ are atoms, h is a datalog relation, and every variable of $\vec{x}$ occurs in $body(\rho) = \{\alpha_1, \ldots, \alpha_m\}$. Abusing notation, we write $\alpha \in \rho$ instead of $\alpha \in body(\rho)$. The variables in $head(\rho)$ are called the *answer variables* of ρ. Given a set of rules P, we let $Dep(P) = (V, E)$ be the directed graph such that: (a) V is the set of all datalog relations occurring in P, and (b) $(p_1, p_2) \in E$ whenever there is a rule $\rho \in P$ where p_1 is the relation in $head(\rho)$, and p_2 occurs in $body(\rho)$. A *non-recursive datalog query* is a pair $Q = (P, q)$ where P is a set of rules such that $Dep(P)$ has no cycle, and q is a datalog relation; its *arity* is the arity of q.

Given a rule ρ and a DL interpretation $\mathcal{I}$, an *assignment* is a function π that maps every variable of ρ to an object in $\Delta^\mathcal{I}$. For a concept atom $A(x) \in \rho$, we write $\mathcal{I} \models_\pi A(x)$ if $\pi(x) \in A^\mathcal{I}$, and for a role atom $r(x_1, x_2) \in \rho$, we write $\mathcal{I} \models_\pi r(x_1, x_2)$ if $(\pi(x_1), \pi(x_2)) \in r^\mathcal{I}$. We call π a *match* for ρ in $\mathcal{I}$ if $\mathcal{I} \models_\pi \alpha$ for all DL-atoms $\alpha \in \rho$.

A tuple $\vec{t}$ is an *answer* to a query $Q = (P, h)$ in an interpretation $\mathcal{I}$ if there exists a rule $\rho = h(\vec{x}) \leftarrow \beta$ in P and a match π for ρ in $\mathcal{I}$ such that (i) $\vec{t} = \pi(\vec{x})$ and (ii) for each non-DL-atom $p(\vec{y}) \in \rho$, $\pi(\vec{y})$ is an answer to (P, p) in $\mathcal{I}$. We use $\mathsf{ans}(Q, \mathcal{I})$ to denote the set of answers to Q in $\mathcal{I}$. The set $\mathsf{cert}(Q, \mathcal{K})$ of *certain answers* to an n-ary query Q over a KB $\mathcal{K}$ is defined as $\{\vec{a} \in (\mathsf{N_I})^n \mid \vec{a}^\mathcal{I} \in \mathsf{ans}(Q, \mathcal{I})$ for any model $\mathcal{I}$ of $\mathcal{K}\}$.

A *conjunctive query (CQ)* is a non-recursive datalog query of the form $(\{\rho\}, q)$, such that $body(\rho)$ contains only DL-atoms. Since the particular relation q is irrelevant, we will use single rules (or rule bodies) to denote CQs.

In this paper, we focus on the decision problem known as the *query output tuple (QOT)* problem, which takes as input a query Q, a KB $(\mathcal{T}, \mathcal{A})$, and a tuple of individuals $\vec{a}$, and consists in deciding whether $\vec{a} \in \mathsf{cert}(Q, (\mathcal{T}, \mathcal{A}))$. Whenever we talk about the *complexity of query answering*, we mean the computational complexity of the QOT problem. We focus on *combined complexity*, which is measured in terms of the size of the whole input ($\vec{a}$, Q, $\mathcal{T}$, $\mathcal{A}$).

Canonical Models Every consistent DL-Lite or $\mathcal{ELH}$ KB $(\mathcal{T}, \mathcal{A})$ possesses a *canonical model* $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. For DL-Lite$_\mathcal{R}$, the domain $\Delta^{\mathcal{T},\mathcal{A}}$ of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ consists of all words $aR_1 \ldots R_n$ ($n \geq 0$) such that $a \in \mathsf{Ind}(\mathcal{A})$, $R_i \in \overline{\mathsf{N_R}}$, and:

- if $n \geq 1$, then $\mathcal{T}, \mathcal{A} \models \exists R_1(a)$;
- for $1 \leq i < n$, $\mathcal{T} \models \exists R_i^- \sqsubseteq \exists R_{i+1}$ and $R_i^- \neq R_{i+1}$.

We say that $w' \in \Delta^{\mathcal{T},\mathcal{A}}$ is a *child* of $w \in \Delta^{\mathcal{T},\mathcal{A}}$ if $w' = wR$ for some R. The interpretation function is defined as follows:

$$\begin{aligned}
a^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} &= a \text{ for all } a \in \mathsf{Ind}(\mathcal{A}) \\
A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} &= \{a \in \mathsf{Ind}(\mathcal{A}) \mid \mathcal{T}, \mathcal{A} \models A(a)\} \\
&\qquad \cup \{aR_1 \ldots R_n \mid n \geq 1 \text{ and } \mathcal{T} \models \exists R_n^- \sqsubseteq A\} \\
r^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} &= \{(a, b) \mid r(a, b) \in \mathcal{A}\} \cup \\
&\qquad \{(w_1, w_2) \mid w_2 = w_1 S \text{ and } \mathcal{T} \models S \sqsubseteq r\} \cup \\
&\qquad \{(w_2, w_1) \mid \ w_2 = w_1 S \text{ and } \mathcal{T} \models S \sqsubseteq r^-\}
\end{aligned}$$

The construction of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ for $\mathcal{ELH}$ KBs is similar, please refer to the appendix for details. Note that for both DL-Lite$_\mathcal{R}$ and $\mathcal{ELH}$ KBs, $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ is composed of a *core*, which is obtained by restricting $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ to the objects in $\mathsf{Ind}(\mathcal{A})$, and an *anonymous part* consisting of (possibly infinite) trees rooted at objects in the core. It is well-known that $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ can be homomorphically mapped into any model of $\mathcal{T}$ and $\mathcal{A}$, yielding:

Fact 1. *Let $\mathcal{K}$ be a consistent DL-Lite or $\mathcal{ELH}$ KB, and let $\mathcal{I}_\mathcal{K}$ be its canonical model. Then $\mathsf{cert}(Q, \mathcal{K}) = \mathsf{ans}(Q, \mathcal{I}_\mathcal{K})$ for every non-recursive datalog query Q.*

3 General Tractability Result

In this section, we observe that for both DL-Lite and $\mathcal{ELH}$, the answers to a CQ ρ over a consistent KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ coincide with the answers to ρ over an interpretation $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ that can be constructed in polynomial time from $\mathcal{K}$ and ρ. Since $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ can be viewed as a relational database, we obtain that any class of CQs that is tractable for plain databases is also tractable for KBs formulated in these DLs.

To establish this result, we rely on the following well-known property of query matches in DL-Lite and $\mathcal{ELH}$ (cf. [Kikot *et al.*, 2012; Lutz *et al.*, 2009]):

Lemma 2. *Consider a consistent DL-Lite or $\mathcal{ELH}$ KB $(\mathcal{T}, \mathcal{A})$, a CQ ρ, and an object w in the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. Then all query matches π for ρ that coincide on the set of variables $\{x \mid \pi(x) = w\}$ coincide also on the sets of variables that are matched to the children of w.*

It follows from Lemma 2 that if there is a query atom $R(x, y)$ such that $\pi(y)$ is a child of $\pi(x)$ in the anonymous part, then it is uniquely determined which other query variables have to be matched inside the tree rooted at $\pi(x)$, and how these variables are ordered into a tree. We can exploit this property to construct the desired $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ in two steps as follows:

(1) First, we generate from ρ a polynomial number of tree-shaped queries, which correspond to the different ways that a subquery of ρ can be mapped inside a tree in the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. Each query is generated by selecting an atom $R(x, y) \in \rho$ and considering what happens if x and y were to be mapped respectively to a node w and one of its children in the anonymous part. By repeatedly applying Lemma 2, we can determine which other variables must be matched inside the tree rooted at w, and how the resulting subquery collapses into a tree.

(2) Only the fragments of the anonymous part of the canonical model into which one of these tree-shaped queries can be homomorphically embedded can participate in query matches. Hence, by appropriately augmenting the core with instantiations of these tree-shaped queries, we obtain an interpretation $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ that is sufficient for retrieving all query answers. Specifically, we attach to each individual a a copy of each tree-shaped query for which there is a match rooted at a. To handle the case of (parts of) queries whose matches may be detached from the core, we also instantiate a disconnected copy of each tree-shaped query which can be mapped inside the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$.

Construction for DL-Lite The following notion of *tree witness* for DL-Lite was defined in [Kontchakov *et al.*, 2010]. Let ρ be a CQ and let $R(x, y)$ be such that either $R(x, y) \in \rho$ or $R^-(y, x) \in \rho$. A *tree witness* for $R(x, y)$ in ρ is a partial map f from the variables in ρ to words over the alphabet $\overline{\mathsf{N_R}}$ such that its domain is minimal (w.r.t. set-theoretic inclusion) and the following conditions hold:

- $f(y) = R$;
- if $f(z) = wS$, $S'(z, z') \in \rho$ or $S'^-(z', z) \in \rho$, and $S' \neq S^-$, then $f(z') = wSS'$; and
- if $f(z) = wS$ and $S(z', z) \in \rho$ or $S^-(z, z') \in \rho$, then $f(z') = w$.

We remark that each tree witness f naturally corresponds to a tree-shaped CQ obtained by restricting the original CQ ρ to the variables in the domain of f and unifying variables z, z' with $f(z) = f(z')$. By definition, there is at most one tree witness for each $R(x, y)$, which we denote by $f_{R(x,y)}$. We say that a tree witness $f_{R(x,y)}$ is *valid w.r.t.* $\mathcal{T}$ if for every word $R_1 \ldots R_n$ in the range of $f_{R(x,y)}$, and every $1 \leq i < n$, we have $R_{i-1} \neq R_i$ and $\mathcal{T} \models \exists R_{i-1}^- \sqsubseteq \exists R_i$. Existence and validity of tree witnesses can be tested in polynomial time.

If a match π for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ maps x and y to objects w and wR respectively, then there is a tree witness $f_{R(x,y)}$ for $R(x, y)$ in ρ which is valid w.r.t. $\mathcal{T}$ and such that $\pi(z) = wf_{R(x,y)}(z)$ for each variable z in the domain of $f_{R(x,y)}$. Moreover, we may assume that if $\pi(x)$ is minimal (that is, there is no $\pi(x')$ which is a prefix of $\pi(x)$), then $\pi(x)$ is within distance $|\mathcal{T}|$ of the ABox. Hence, the matches for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ coincide with the matches of ρ in the structure $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ obtained by adding to the core:

(a) the objects aw such that w occurs in the range of a valid tree witness $f_{R(x,y)}$ and $\mathcal{T}, \mathcal{A} \models \exists R(a)$.

(b) the objects xSw where x is a variable in ρ and S, w satisfy: (i) there is an individual $a \in \mathsf{Ind}(\mathcal{A})$ and a chain of (possibly inverse) roles $R_1, \ldots, R_n$ of length at most $|\mathcal{T}|$ such that $R_n = S$, $\mathcal{T}, \mathcal{A} \models \exists R_1(a)$, and for each $1 < i \leq n$, $R_{i-1}^- \neq R_i$ and $\mathcal{T} \models \exists R_{i-1}^- \sqsubseteq \exists R_i$, and (ii) there exists a variable y and valid tree witness $f_{R(x,y)}$ with $S^- \neq R$ and $\mathcal{T} \models \exists S^- \sqsubseteq \exists R$ whose range contains w. Note that x is *not* an object of the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$.

To extend the interpretations of concept and role names to these new objects, we let $wR \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ whenever $\mathcal{T} \models \exists R^- \sqsubseteq A$, and $(w, wR) \in R^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ for new objects w, wR.

We remark that the existence of a role chain and individual having the properties stated in (b)(i) can be decided in polynomial time by initializing a set Reach with all roles S such that $\mathcal{T}, \mathcal{A} \models \exists S(a)$ for some $a \in \mathsf{Ind}(\mathcal{A})$, and then adding U to Reach whenever there is $V \in \mathsf{Reach}$ such that $\mathcal{T} \models V^- \sqsubseteq U$. Since there are only polynomially many objects of the forms aw and xSw as above, and instance checking and TBox reasoning are tractable for DL-Lite KBs [Calvanese *et al.*, 2007], it follows that $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ can be constructed in polynomial time.

Construction for $\mathcal{ELH}$ In what follows, it will prove convenient to use conjunction as a role constructor: if $r_1, r_2 \in \mathsf{N_R}$ are role names, then their conjunction $r_1 \sqcap r_2$ is a role whose interpretation is $(r_1 \sqcap r_2)^\mathcal{I} = r_1{}^\mathcal{I} \cap r_2{}^\mathcal{I}$. We denote by $\mathcal{ELH}_\sqcap$ the extension of $\mathcal{ELH}$ with role conjunction.

We introduce a notion of tree witness for $\mathcal{ELH}$, inspired by the fork elimination procedure from [Lutz, 2008]. Let ρ be a CQ and $\alpha = r(x, y) \in \rho$. We begin by defining a set D_α and equivalence relation $\sim_\alpha$ over D_α by initializing D_α to $\{x, y\}$ and $\sim_\alpha$ to the trivial equivalence relation, and then applying the following rules until convergence:

- if $s(z, z') \in \rho$, $z \in D_\alpha$, and $z \not\sim_\alpha x$, then add z' to D_α
- if $s(u, u') \in \rho$, $t(z, z') \in \rho$, $u, u', z' \in D_\alpha$, and $u' \sim_\alpha z'$, then add z to D_α and put $u \sim_\alpha z$

Note that D_α and $\sim_\alpha$ are uniquely defined and can be computed in polynomial time in the size of ρ. We let ρ_α be the query obtained by restricting ρ to the variables in D_α, then replacing each variable z by its equivalence class $[z]$ under $\sim_\alpha$. We define a directed graph G_α whose nodes are the equivalence classes in $\sim_\alpha$ and whose edges are the atoms in ρ_α. If G_α contains no (directed) cycles, then we define the *tree witness* for $\alpha = r(x, y)$ in ρ as the map $f : D_\alpha \to (2^{\mathsf{N_R}} \times 2^{\mathsf{N_C}})^*$ with:

- $f(z) = \varepsilon$ if $z \sim_\alpha x$
- $f(z) = M_1 N_1 \dots M_k N_k$ if $[u_0], \dots, [u_k]$ is the unique path in G_α with $u_0 \sim_\alpha x$ and $u_k \sim_\alpha z$, and for every $1 \leq i \leq k$, we have $M_i = \{s \mid s([u_{i-1}], [u_i]) \in G_\alpha)$ and $N_i = \{A \mid A([u_i]) \in G_\alpha\}$

To every tree witness f, we can naturally associate an $\mathcal{ELH}_\sqcap$ concept $\mathsf{conc}_f(\varepsilon)$ as follows: if $f(z) = w$ is a leaf, we let $\mathsf{conc}_f(w) = \top$, and if $f(z) = w$ has children $w_1, \dots, w_n$ with $w_i = wM_iN_i$, then $\mathsf{conc}_f(w) = \bigsqcap_{i=1}^n \exists(\bigsqcap_{r \in M_i} r).(\bigsqcap_{A \in N_i} A \sqcap \mathsf{conc}_f(w_i))$. It follows from [Rudolph *et al.*, 2008] that it can be checked in polynomial time whether $\mathcal{T}, \mathcal{A} \models \mathsf{conc}_f(\varepsilon)(a)$. Using a reachability construction similar to the one for DL-Lite, we can also test in polynomial time whether $\mathsf{conc}_f(\varepsilon)$ is non-empty in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$.

We are now ready to define the interpretation $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Its domain is the set of words consisting of each individual in $\mathcal{A}$, each $a\alpha w$ such that $w \neq \varepsilon$ is in the range of a tree witness f for α in ρ such that $(\mathcal{T}, \mathcal{A}) \models \mathsf{conc}_f(\varepsilon)(a)$, each $B\alpha w$ such that $w \neq \varepsilon$ is in the range of a tree witness f for α in ρ and $\mathsf{conc}_f(\varepsilon)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ contains an object of the form $w'B$, and each B for which some $B\alpha w$ is present. Concept and role names are interpreted as follows:

$$\begin{aligned} A^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}} =& \{a \in \mathsf{Ind}(\mathcal{A}) \mid \mathcal{T}, \mathcal{A} \models A(a)\} \cup \\ & \{B \mid \mathcal{T} \models B \sqsubseteq A\} \cup \\ & \{wMN \mid \mathcal{T} \models \bigsqcap_{B \in N} B \sqsubseteq A\} \\ r^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}} =& \{(a, b) \mid \mathcal{T}, \mathcal{A} \models r(a, b)\} \cup \\ & \{(\sigma, \sigma\alpha MN) \mid \mathcal{T} \models s \sqsubseteq r \text{ for some } s \in M\} \cup \\ & \{(w, wMN) \mid \mathcal{T} \models s \sqsubseteq r \text{ for some } s \in M\} \end{aligned}$$

and each individual a is interpreted as itself ($a^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}} = a$).

The following theorem resumes the key properties of the interpretations $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ just described.

Theorem 3. *Let ρ be a CQ, let $(\mathcal{T}, \mathcal{A})$ be a consistent DL-Lite or $\mathcal{ELH}$ knowledge base, and let $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ be the interpretation defined previously. The following statements hold:*

1. *$\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ can be built in polynomial time in ρ, $\mathcal{T}$ and $\mathcal{A}$.*
2. *For every tuple $\vec{a}$ of individuals, $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A}})$ iff $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A},\rho})$.*

In light of Theorem 3 and Fact 1, to determine whether $\vec{a} \in \mathsf{cert}(\rho, (\mathcal{T}, \mathcal{A}))$, it is sufficient to test the consistency of $(\mathcal{T}, \mathcal{A})$ and then, if $(\mathcal{T}, \mathcal{A})$ is consistent, to decide whether $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A},\rho})$. If we view $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ as a plain relational database, the latter check is just a special case of the QOT problem. Hence, we obtain the desired result:

Corollary 4. *Let $\mathcal{Q}$ be a class of CQs for which the query output tuple problem over relational databases is decidable in polynomial time. Then the query output tuple problem for $\mathcal{Q}$ is also tractable for KBs formulated in DL-Lite and $\mathcal{ELH}$.*

The construction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ above is easily extended to DL-Lite$_\mathcal{R}$ using the notions of canonical models and tree witnesses from [Kikot *et al.*, 2012]. However, the construction is no longer polynomial since there can be exponentially many tree witnesses for a single atom $R(x, y)$ in ρ [Kikot *et al.*, 2011], and so we do not obtain an analogue of Theorem 3. Indeed, it follows from results by Kikot et al. 2011 that the QOT problem for tree-shaped CQs is NP-complete for DL-Lite$_\mathcal{R}$ KBs. Therefore, to obtain tractability results to DL-Lite$_\mathcal{R}$, one must impose some syntactic restriction on $\mathcal{T}$ and ρ that ensures a polynomial number of tree witnesses, e.g. the absence of *twisty roles* proposed in [Kikot *et al.*, 2012].

4 Answering Acyclic Queries

The expansion technique presented in Section 3 allows us to convert any polynomial-time algorithm for evaluating a tractable class of CQs over relational databases into a polynomial-time algorithm for evaluating the same class of queries over DL-Lite and $\mathcal{ELH}$ KBs. However, the resulting algorithm would involve building the structure $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ for each input query ρ, which is clearly undesirable. We now present our main contribution: a polynomial-time procedure for evaluating acyclic CQs which is based upon rewriting CQs into non-recursive datalog programs. We present the approach for DL-Lite KBs, but discuss at the end of the section how the approach can be adapted to DL-Lite$_\mathcal{R}$ and $\mathcal{ELH}$.

We begin with some preliminaries. As usual, the query graph $G(\rho)$ of a CQ ρ is defined as the undirected graph whose nodes are the variables of ρ, and that has an edge between x and x' if ρ contains a (body) atom $r(x, x')$ or $r(x', x)$. A CQ ρ is *acyclic* if $G(\rho)$ is acyclic, and *rooted* if every connected component of $G(\rho)$ has at least one answer variable. We consider a slight generalization of acyclicity: we say that a CQ ρ is *acyclic modulo answer variables* (*a-acyclic* for short) if the graph $G^-(\rho)$ obtained by deleting from $G(\rho)$ each edge (x, x'), where x, x' are answer variables is acyclic.

Our rewriting procedure works on queries which are both rooted and a-acyclic (we discuss later the non-rooted case). To every rooted a-acyclic CQ ρ, we associate the set of connected components $\{T_1, \dots, T_n\}$ of $G^-(\rho)$. Because ρ is rooted, every T_i is a connected acyclic graph containing at least one vertex which is an answer variable. We select an arbitrary answer variable x_i for each T_i and designate it as the root of T_i, allowing us to view T_i as a tree. Then, given a pair of variables x, y of ρ, we call y a *child* of x if y is a child of x in the (unique) tree T_i that contains x, and define *descendant* as the transitive closure of child. For a variable x of ρ, we denote by $\vec{x}_\rho$ (resp. $\vec{x}_\rho^+$) the tuple consisting of all answer variables which are descendants of x (resp. which are among x and its descendants).

Rewriting procedure for rooted queries Consider a rooted a-acyclic CQ $\rho = q(\vec{x}) \leftarrow \alpha$ and a DL-Lite TBox $\mathcal{T}$, and let X be the set of answer variables which are roots in ρ (see previously). We rewrite the query ρ into the non-recursive datalog program $\mathsf{rew}_\mathcal{T}(\rho) = (P, q)$ defined as follows. In addition to q, the program uses the following datalog relations: (i) $(|\vec{x}_\rho|+1)$-ary relations q_x, q'_x for every variable x of ρ, and (ii) unary relations q_A and $q_{\exists R}$ for every $A \in \mathsf{N_C}$ and $R \in \overline{\mathsf{N_R}}$

occurring in ρ. We now describe the rules in P. There is a single top-level rule defining q:

$$q(\vec{x}) \leftarrow \bigwedge_{x \in X} q_x(x, \vec{x}_\rho) \wedge \bigwedge_{x_i, x_j \in \vec{x},\, r(x_i, x_j) \in \rho} r(x_i, x_j) \qquad (1)$$

For every variable x in ρ, with $Y = \{y_1, \dots, y_n\}$ the set of children of x in ρ, we have the following rule

$$q_x(x, \vec{x}_\rho) \leftarrow \bigwedge_{A(x) \in \rho} q_A(x) \wedge \bigwedge_{r(x,x) \in \rho} r(x, x) \wedge \bigwedge_{y \in Y} q'_y(x, \vec{y}^{+}_{\rho}) \quad (2)$$

and for every $y \in Y$, we also have

$$q'_y(x, \vec{y}^{+}_{\rho}) \leftarrow \bigwedge_{r(x,y) \in \rho} r(x, y) \wedge \bigwedge_{s(y,x) \in \rho} s(y, x) \wedge q_y(y, \vec{y}_\rho) \quad (3)$$

and for all $y \in Y$ satisfying the following conditions:

(i) there is an atom $R(x, y) \in \rho$ or $R^-(y, x) \in \rho$ and the tree witness $f_{R(x,y)}$ exists and is valid

(ii) for every u in domain of $f_{R(x,y)}$ with $f_{R(x,y)}(u) = wS$ and $A(u) \in \rho$, we have $\mathcal{T} \models \exists S^- \sqsubseteq A$

(iii) the set $Z = \{z \mid f_{R(x,y)}(z) = \varepsilon \wedge z \neq x\}$ contains all answer variables in the domain of $f_{R(x,y)}$

we additionally have the rule

$$q'_y(x, \vec{u}) \leftarrow q_{\exists R}(x) \wedge \bigwedge_{z \in Z} q_z(x, \vec{z}_\rho) \qquad (4)$$

where $\vec{u}$ is obtained from $\vec{y}_\rho$ by replacing each $z \in Z$ by x. Finally, for every $B \in \mathsf{N_C} \cup \{\exists R \mid R \in \overline{\mathsf{N_R}}\}$ with q_B a datalog relation in P, we have the rules

$$\begin{array}{ll} q_B(x) \leftarrow A(x) & \text{for all } A \in \mathsf{N_C} \text{ such that } \mathcal{T} \models A \sqsubseteq B \\ q_B(x) \leftarrow s(x, y) & \text{for all } s \in \mathsf{N_R} \text{ such that } \mathcal{T} \models \exists s \sqsubseteq B \quad (5) \\ q_B(x) \leftarrow s(y, x) & \text{for all } s \in \mathsf{N_R} \text{ such that } \mathcal{T} \models \exists s^- \sqsubseteq B \end{array}$$

Intuitively, the relation q_x corresponds to the query $\rho|x$ whose answer variables are $\{x\} \cup \vec{x}_\rho$ and whose body is obtained by restricting the body of ρ to the atoms whose arguments among x and its descendants; whereas the relation q'_y corresponds to the query $\rho|xy$ (with y a child of x) obtained by adding to $\rho|y$ the role atoms linking x and y. Rule (1) stipulates that a tuple is in the answer to ρ if it makes true all of the role atoms linking two answer variables and each of the queries $\rho|x$ associated with a root variable x of ρ. Then rule (2) states that to make $\rho|x$ hold at an individual, we must satisfy the concept atoms for x and the query $\rho|xy$ for each child y of x. Rules (3) and (4) provide two ways of satisfying $\rho|xy$. The first way, captured by rule (3), is to map y to an ABox individual, in which case the role atoms between x and y must occur in the ABox, and the query $\rho|y$ must hold at this individual. The second possibility, treated by rule (4), is that y is mapped to an element of the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ which is a child of x. For this to occur, several conditions must be verified. First, if y is an R-successor of x, then the tree witness $f_{R(x,y)}$ must exist and be valid w.r.t. $\mathcal{T}$. Second, we must ensure that all concept atoms concerning variables that are mapped inside the anonymous part by $f_{R(x,y)}$ are satisfied (this is checked in item (ii)). Finally, since answer variables cannot be mapped inside the anonymous part, we need condition (iii), which checks that every answer variable z in the domain of $f_{R(x,y)}$ is such that $f_{R(x,y)}(z) = \varepsilon$. If all of these conditions are met, then rule (4) states that the query $\rho|xy$ can be satisfied by making $\exists R$ hold at x (thereby guaranteeing the existence of the required paths in the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$) *and* by satisfying the remainder of the query $\rho|xy$ (i.e. the query obtained by removing the atoms mapped inside the anonymous part). The latter corresponds precisely to the union of the queries $\rho|z$ where z is a descendant of x with $f_{R(x,y)}(z) = \varepsilon$. Finally, the rules in (5) provide the standard rewriting of concepts A and $\exists R$ w.r.t. $\mathcal{T}$.

We establish the correctness of our rewriting procedure.

Theorem 5. *Let ρ be a rooted a-acyclic CQ and $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ a DL-Lite KB. Then* $\mathsf{cert}(\rho, \mathcal{K}) = \mathsf{ans}(\mathsf{rew}_{\mathcal{T}}(\rho), \mathcal{I}_{\mathcal{A}})$.

Proof idea. The key step in the proof is showing that for every variable x in ρ, $\mathsf{cert}(\rho|x, \mathcal{K}) = \mathsf{ans}((P, q_x), \mathcal{I}_{\mathcal{A}})$. This can be proven by induction on the number of variables in $\rho|x$, utilizing Fact 1 and properties of tree witnesses. □

The next theorem shows that our rewriting procedure provides a polynomial-time algorithm for evaluating rooted a-acyclic conjunctive queries.

Theorem 6. *Given a rooted a-acyclic CQ ρ and a DL-Lite KB $(\mathcal{T}, \mathcal{A})$, the program $\mathsf{rew}_{\mathcal{T}}(\rho)$ can be computed in polynomial time in the size of ρ and $\mathcal{T}$, and $\vec{a} \in \mathsf{ans}(\mathsf{rew}_{\mathcal{T}}(\rho), \mathcal{I}_{\mathcal{A}})$ can be tested in polynomial time in the size of $\mathsf{rew}_{\mathcal{T}}(\rho)$ and $\mathcal{A}$.*

Proof. For the first point, we observe that the number of relations in $\mathsf{rew}_{\mathcal{T}}(\rho)$ is linear in the number of atoms in ρ and that each relation is defined using linearly many rules in the size of ρ and $\mathcal{T}$. We also note that testing the conditions for rules of type (4) can be done in polynomial time in the size of ρ and $\mathcal{T}$ (cf. Section 3). The second statement is true because once the answer variables in $\mathsf{rew}_{\mathcal{T}}(\rho)$ have been instantiated with the individuals in $\vec{a}$, we have a non-recursive datalog program (with constants) where every rule contains at most two variables. It is known that datalog programs of this form can be evaluated in polynomial time (cf. [Dantsin *et al.*, 2001]). □

Handling non-rooted queries. We now return to the case of non-rooted a-acyclic CQs, and show that such queries can be answered via a reduction to the rooted query case. Given an a-acyclic CQ ρ over a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, the set $\mathsf{cert}(\rho, \mathcal{K})$ can be computed using the following steps:

1. If ρ is rooted, then return $\mathsf{cert}(\rho, \mathcal{K})$.
2. Choose a maximally connected component β of ρ, such that β contains no answer variable.
3. If β has a match fully in the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, then drop β from ρ and go to step 1.
4. If there is a variable x in β such that the rooted query $g(x) \leftarrow \beta$ has a non-empty answer over $\mathcal{K}$, then drop β from ρ and go to step 1. Otherwise, return $\emptyset$.

It is not hard to show that β satisfies the condition in step 3 just in the case that there exists a variable x and a role $S \in \overline{\mathsf{N_R}}$ which is reachable in the canonical model (cf. item (b)(i) of the construction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$) such that there is a valid tree witness f for $S(z,x)$ in $\beta_{S,x} = \beta \cup \{S(z,x)\}$ (z fresh) with:

(i) if $f(y) = \varepsilon$, then $y = z$, and

(ii) $\mathcal{T} \models \exists R^- \sqsubseteq A$ whenever $f(y) = wR$ and $A(y) \in \beta$.

It follows that step 3 can be carried out in polynomial time, and thus we obtain a polynomial-time procedure for solving the QOT problem for arbitrary a-acyclic CQs.

Adapting the rewriting for DL-Lite$_\mathcal{R}$ and $\mathcal{ELH}$ We now discuss how to modify the rewriting to handle DL-Lite$_\mathcal{R}$ and $\mathcal{ELH}$ KBs. First, since both DLs support role inclusions, we must replace atoms $r(x,y)$ by atoms $q_r(x,y)$, and add the corresponding rules $q_r(x,y) \leftarrow S(x,y)$ with $\mathcal{T} \models S \sqsubseteq r$. Then the algorithm can be directly employed for DL-Lite$_\mathcal{R}$, but using the notion of tree witness in [Kikot *et al.*, 2012]. Similarly as in Section 3, the algorithm may not be polynomial since there can be exponentially many tree witnesses.

For $\mathcal{ELH}$, apart from handling role inclusions as above, rule (4) must be modified to use the $\mathcal{ELH}$ version of tree witnesses. In particular, instead of an atom $q_{\exists R}(x)$, we use $q_{C_f}(x)$, where C_f is the concept induced by the tree witness f for $r(x,y)$. Note that there are only linearly many tree witnesses, hence only linearly many new relations q_{C_f}. Assuming that the $\mathcal{ELH}$ TBox is in normal form (cf. [Baader *et al.*, 2005]), we only need to consider atomic concepts A which entail C, yielding a linear number of rules of the form $q_{C_f}(x) \leftarrow q_A(x)$. Finally, we must modify the rules in (5) defining the relations q_A to capture entailment in $\mathcal{ELH}$, which may require the use of recursive datalog rules (see e.g. [Eiter *et al.*, 2012]). Importantly, these rules have at most two variables each, and so they do not impact the polynomial upper bound argument. Thus, by using this modified rewriting, and handling non-rooted CQs in a similar way to DL-Lite, we obtain a polynomial-time algorithm for deciding the QOT problem for a-acyclic CQs over $\mathcal{ELH}$ KBs.

5 Preliminary Evaluation

We developed a prototype rewriting system that takes as input a rooted acyclic CQ ρ and a DL-Lite$_\mathcal{R}$ TBox $\mathcal{T}$, and outputs an SQL statement expressing the resulting non-recursive datalog program $\mathsf{rew}_\mathcal{T}(\rho)$ (using common table expressions). We evaluated the result over ABoxes stored in a relational database, using the PostgreSQL database system, and Owlgres [Stocker and Smith, 2008] for loading the data.

To test our prototype, we used the LUBM$^\exists_{20}$ ontology described in [Lutz *et al.*, 2012], which adds concept inclusions with additional concept names, and with existential concepts on the right hand side, to the original LUBM ontology [Guo *et al.*, 2005]. We considered three acyclic queries from the benchmark in [Lutz *et al.*, 2012] (q_2, q_4, and q_5 from the 6 provided queries), which are rather small (at most 4 atoms), and created three additional large acyclic queries, with 13 to 34 atoms, and 7 to 17 variables (q_7, q_8, and q_9). We note that the new queries are also significantly larger than the ones of the REQUIEM test suite ($\leq$ 7 atoms). The importance of handling such larger queries in practice has been previously argued in [Rosati and Almatelli, 2010].

#Uni	q_2	q_4	q_5	q_7	q_8	q_9
20	2.5	3.0	4.2	1.5	1.0	0.0
50	9.0	7.3	4.6	2.0	3.3	0.0
100	20.5	15.0	9.4	4.2	7.2	0.0
150	25.6	21.8	14.1	6.6	11.6	15.2
200	33.5	>600	27.0	15.2	26.9	31.2

Table 1: Scalability of our system (runtime in seconds)

We compared our rewriting procedure with REQUIEM [Pérez-Urbina *et al.*, 2009] and IQAROS [Venetis *et al.*, 2012] which, like most of the existing query rewriting systems for the DL-Lite family, generate unions of CQs rather that non-recursive datalog programs. For the three large queries, both REQUIEM and IQAROS did not terminate (within ten minutes). Even for the small q_4 and q_5, the generated rewritings were too large to be posed directly to an off-the-shelf RDBMs: REQUIEM generated tens of thousands of queries for both, and IQAROS almost 15 thousand for q_4, and almost one thousand for q_5. In contrast, for our approach, the rewriting times were negligible for all queries (under half a second). The rewritings produced by our approach have less than 30 rules for all queries, disregarding the rules (5) of the algorithm (since the latter are independent of the query, we computed them separately and stored them using a database view per concept/role name).

We also tested the feasibility of evaluating our rewritings over large ABoxes. For this, we used the modified LUBM data generator [Lutz *et al.*, 2012] (with 5% incompleteness). Each university has approximately 17k individuals, 28K concept assertions, and 47K role assertions. We carried out our experiments on ABoxes with 20 – 200 universities, resulting in very large ABoxes (up to ca. 1.5 GB on disk). The results reported in Table 1 show that the algorithm scales well.

Finally, we note that a performance comparison with PRESTO [Rosati and Almatelli, 2010], which also outputs non-recursive datalog programs, was not possible because this system is not publicly available. However, we can observe that its underlying algorithm may produce exponential-size rewritings for acyclic CQs, as witnessed by the family of queries $q(x) \leftarrow r(x,y) \wedge \bigwedge_{0 \leq i \leq n} p(y, z_i) \wedge p(u_i, z_i) \wedge B_i(u_i)$ coupled with e.g. the empty TBox[2]. Intuitively, the PRESTO algorithm generates a separate rule for every possible way to select a collection of variables from $\{u_1, \ldots, u_n\}$ and identify them with y. This exponential blow-up suggests that our positive results are not merely an artifact of the datalog representation, but derive also from acyclicity.

6 Future Work

We plan to generalize our rewriting technique to larger tractable classes of CQs, like bounded treewidth CQs. Another direction is to identify suitable restrictions for more expressive DLs that allow for tractable answering of acyclic

[2] In fact, the exponential blowup occurs even without the atoms $B_i(u_i)$, but some quite obvious modifications to the algorithm would resolve the issue. With these atoms present, it appears non-trivial changes to the algorithm would be required.

queries. Our proof-of-concept implementation raises hopes that efficient evaluation of large acyclic queries is feasible, but many challenges must still be addressed. For example, we observe that breaking down the queries into small rules as is done by our rewriting may lead to a loss of structure that could be used by database management systems for optimized evaluation. There are many other aspects, not directly related to the rewriting technique, that must also be taken into account to achieve practicable query answering, such as exploring more efficient forms of representing data in ABoxes, using different kinds of indexes, and considering different translations of our programs into SQL. Using semantic indexes [Rodriguez-Muro and Calvanese, 2012] for handling the rules of type (5) appears particularly promising.

Acknowledgements The authors were supported by a Université Paris-Sud Attractivité starting grant and ANR project PAGODA ANR-12-JS02-007-01 (Bienvenu), FWF project T515-N23 (Ortiz), FWF project P25518-N23 and WWTF project ICT12-015 (Šimkus), Vienna PhD School of informatics and EU project Optique FP7-318338 (Xiao).

References

[Baader *et al.*, 2005] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the EL envelope. In *Proc. of IJCAI*, pages 364–369, 2005.

[Calvanese *et al.*, 2007] 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. *J. Automated Reasoning*, 39(3):385–429, 2007.

[Chekuri and Rajaraman, 1997] Chandra Chekuri and Anand Rajaraman. Conjunctive query containment revisited. In *Proc. of ICDT*, pages 56–70. Springer, 1997.

[Dantsin *et al.*, 2001] Evgeny Dantsin, Thomas Eiter, Georg Gottlob, and Andrei Voronkov. Complexity and expressive power of logic programming. *ACM Computing Survey*, 33(3):374–425, September 2001.

[Eiter *et al.*, 2012] Thomas Eiter, Magdalena Ortiz, Mantas Simkus, Trung-Kien Tran, and Guohui Xiao. Query rewriting for Horn-SHIQ plus rules. In *Proc. of AAAI*. AAAI Press, 2012.

[Gottlob *et al.*, 1999] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proc. of PODS*, pages 21–32. ACM Press, 1999.

[Guo *et al.*, 2005] Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin. LUBM: A benchmark for OWL knowledge base systems. *J. Web Semantics*, 3(2-3):158–182, 2005.

[Kikot *et al.*, 2011] Stanislav Kikot, Roman Kontchakov, and Michael Zakharyaschev. On (in)tractability of OBDA with OWL 2 QL. In *Proc. of DL*. CEUR-WS.org, 2011.

[Kikot *et al.*, 2012] Stanislav Kikot, Roman Kontchakov, and Michael Zakharyaschev. Conjunctive query answering with OWL 2 QL. In *Proc. of KR*. AAAI Press, 2012.

[Kontchakov *et al.*, 2010] Roman Kontchakov, Carsten Lutz, David Toman, Frank Wolter, and Michael Zakharyaschev. The combined approach to query answering in DL-Lite. In *Proc. of KR*. AAAI Press, 2010.

[Levy and Rousset, 1998] Alon Y. Levy and Marie-Christine Rousset. Combining Horn rules and description logics in CARIN. *Artificial Intelligence*, 104(1-2):165–209, 1998.

[Lutz *et al.*, 2009] Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic EL using a relational database system. In *Proc. of IJCAI*, pages 2070–2075, 2009.

[Lutz *et al.*, 2012] Carsten Lutz, Inanc Seylan, David Toman, and Frank Wolter. The combined approach to OBDA: Taming role hierarchies using filters (with appendix). In *Proc. of SSWS+HPCSW*, 2012.

[Lutz, 2008] Carsten Lutz. The complexity of conjunctive query answering in expressive description logics. In *Proc. of IJCAR*, pages 179–193. Springer, 2008.

[Ortiz and Simkus, 2012] Magdalena Ortiz and Mantas Simkus. Reasoning and query answering in description logics. In *Reasoning Web*, pages 1–53. Springer, 2012.

[OWL Working Group, 2009] W3C OWL Working Group. *OWL 2 Web Ontology Language: Document Overview*. W3C Recommendation, 2009. Available at `http://www.w3.org/TR/owl2-overview/`.

[Pérez-Urbina *et al.*, 2009] Héctor Pérez-Urbina, Boris Motik, and Ian Horrocks. A comparison of query rewriting techniques for DL-Lite. In *Proc. of DL*. CEUR-WS.org, 2009.

[Picalausa and Vansummeren, 2011] François Picalausa and Stijn Vansummeren. What are real SPARQL queries like? In *Proc. of SWIM*. ACM, 2011.

[Rodriguez-Muro and Calvanese, 2012] Mariano Rodriguez-Muro and Diego Calvanese. High performance query answering over DL-Lite ontologies. In *Proc. of KR*, pages 308–318. AAAI Press, 2012.

[Rosati and Almatelli, 2010] Riccardo Rosati and Alessandro Almatelli. Improving query answering over DL-Lite ontologies. In *Proc. of KR*. AAAI Press, 2010.

[Rudolph *et al.*, 2008] Sebastian Rudolph, Markus Krötzsch, and Pascal Hitzler. Cheap boolean role constructors for description logics. In *Proc. of JELIA*, pages 362–374, 2008.

[Stocker and Smith, 2008] Markus Stocker and Michael Smith. Owlgres: A scalable OWL reasoner. In *Proc. of OWLED*. CEUR-WS.org, 2008.

[Venetis *et al.*, 2012] Tassos Venetis, Giorgos Stoilos, and Giorgos B. Stamou. Incremental query rewriting for OWL 2 QL. In *Proc. of DL*. CEUR-WS.org, 2012.

[Yannakakis, 1981] Mihalis Yannakakis. Algorithms for acyclic database schemes. In *Proc. of VLDB*, pages 82–94. IEEE Computer Society, 1981.

A Proof of Theorem 3

The arguments underlying the first statement of Theorem 3 (polynomial-time construction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$) were already given in the text, so we concentrate here on proving the second statement of Theorem 3. The proof is split into two parts, with Propositions 7 handling DL-Lite KBs and Proposition 12 treating $\mathcal{ELH}$ KBs.

We start by giving the proof for DL-Lite.

Proposition 7. *Let ρ be a CQ, let $(\mathcal{T},\mathcal{A})$ be a consistent DL-Lite knowledge base, and let $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ be the interpretation defined in Section 3. Then for every tuple $\vec{a}$ of individuals, $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A}})$ iff $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A},\rho})$.*

Proof. Let $\vec{x}$ be the tuple of answer variables of ρ. First suppose that $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A}})$. By Fact 1, there exists a match π for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\pi(\vec{x}) = \vec{a}$. Let $\rho_1, \ldots, \rho_n$ be the maximally connected components of ρ. For each ρ_i, there are two possibilities:

- ρ_i contains a variable v such that $\pi(v) \in \mathsf{Ind}(\mathcal{A})$
- all variables in ρ_i are mapped by π to the anonymous part.

First consider the case in which ρ_i contains v such that $\pi(v) \in \mathsf{Ind}(\mathcal{A})$. We claim that in this case, the assignment π is a match for ρ_i in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Because the interpretation of concept and role names is identical in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ and $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ for elements common to both domains, it is suffices to show that every object $\pi(u)$ (with u a variable in ρ_i) belongs to the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. This is trivially the case when $\pi(u) \in \mathsf{Ind}(\mathcal{A})$, since the core of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ is part of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Next consider the case in which $\pi(u) = aR_1 \ldots R_\ell$. The presence of $aR_1 \ldots R_\ell$ in the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ implies that $\mathcal{T},\mathcal{A} \models \exists R_1(a)$. Because ρ_i is connected and $\pi(v) \in \mathsf{Ind}(\mathcal{A})$, there must exist a sequence of variables $u_0, \ldots, u_\ell$ with $u_\ell = u$ such that:

- $\pi(u_0) = a$ and $\pi(u_1) = aR_1$
- for every $1 \leq j \leq \ell$, there is a role atom in ρ_i which contains both u_{j_1} and u_j
- for every $1 \leq j \leq \ell$, $\pi(u_j) \neq a$

From the first two items and the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, we can infer that either $R_1(u_0, u_1)$ or $R_1^-(u_1, u_0)$ appears in ρ_i. It follows from the properties of tree witnesses that the tree witness $f_{R_1(u_0,u_1)}$ exists and is valid. By the second and third items, the word $R_1 \ldots R_\ell$ belongs to the range of $f_{R_1(u_0,u_1)}$. Thus, by point (i) of the construction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$, the object $\pi(u) = aR_1 \ldots R_\ell$ is in the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$.

Now consider the case in which every variable u in ρ_i is mapped to the anonymous part. Since ρ_i is connected, and the anonymous part has a forest structure, we can find a variable v such that $\pi(v)$ is a prefix of $\pi(u)$ for every u in ρ_i. Let $a \in \mathsf{Ind}(\mathcal{A})$, $S \in \overline{\mathsf{N_R}}$, and $w \in \overline{\mathsf{N_R}}^*$ be such that $\pi(v) = awS$. Define an assignment μ to the variables of ρ_i by setting $\mu(u) = vSw$ where w is the unique word such that $\pi(u) = \pi(v)w$. Our aim is to show that μ is a match for ρ_i in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Specifically, we need to show (1) that all objects in the image of μ belong to the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$, and (2) all atoms in ρ_i are made true by μ. For (1), observe that the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ ensures that if $bw_1Rw_2Rw_3$ is an element in the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, then bw_1Rw_3 also belongs to the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. Thus, by repeatedly deleting subwords from $\pi(v) = wS$, we obtain a word $aR_1 \ldots R_n$ in the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $R_n = S$ and $n \leq |\overline{\mathsf{N_R}}| < |\mathcal{T}|$. The presence of $aR_1 \ldots R_n$ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ implies that $\mathcal{T},\mathcal{A} \models \exists R_1(a)$ and for each $1 < i \leq n$, $R_{i-1}^- \neq R_i$ and $\mathcal{T} \models \exists R_{i-1}^- \sqsubseteq \exists R_i$. We have thus shown that condition (b)(i) from the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ holds for words of the form vSw, and it remains to establish (b)(ii). Let $\alpha_1, \ldots, \alpha_\ell$ be all the role atoms involving the variable v. If the α_j takes the form $R(v, v')$ or $R^-(v', v)$, then we must have $\pi(v') = \pi(v)R$, so the tree witness $f_j = f_{R(v,v')}$ exists and is valid. Moreover, since ρ_i is connected, every variable u must belong to the domain of one of the tree witnesses f_j. Now suppose that $\mu(u) = vSw$ and u belongs to the domain of f_j. Then we have $\pi(u) = \pi(v)w$, and so $f_j(u) = w$ and condition (b)(ii) holds for vSw. To show (2), first remark that if $A(u) \in \rho_i$, then we must have $\pi(u) \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$, and so $\pi(u)$ must end with a symbol R such that $\mathcal{T} \models \exists R^- \sqsubseteq A$. By construction, $\mu(u)$ also ends with R, and so the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ yields $\mu(u) \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. Now consider a role atom $t(u, u') \in \rho_i$. From the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ and the fact that π is a match, we know that either $\pi(u') = \pi(u)t$ or $\pi(u) = \pi(u')t^-$. It follows that either $\mu(u') = \mu(u)t$ or $\mu(u) = \mu(u')t^-$. In both cases, we obtain $(\mu(u), \mu(u')) \in t^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$. This completes our proof of (2) and establishes that μ is a match for ρ_i in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$.

We have thus shown that every query ρ_i corresponding to a maximally connected component of ρ has a match in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$, and moreover, any answer variables are sent to the corresponding individual from $\vec{a}$. As the queries ρ_i are on disjoint variables, we can combine the matches for the different ρ_i to obtain a match for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ which sends $\vec{x}$ to $\vec{a}$. From this, we can conclude that $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A},\rho})$.

For the other direction, let $h : \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}} \to \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ be the mapping defined as follows:

- if $o = aw$ with $a \in \mathsf{Ind}$, then $h(o) = o$
- if $o = xSw$ with $x \notin \mathsf{Ind}$, then $h(o) = aR_1 \ldots R_n w$ where a and $R_1 \ldots R_n$ are chosen so as to satisfy condition (b)(i)

It is straightforward to verify that h is a homomorphism from $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ to $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. As certain answers to conjunctive queries are preserved under homomorphisms, $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A},\rho})$ implies $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A}})$. □

Before proceeding to the proof of Proposition 12, we must recall some notions and terminology concerning $\mathcal{ELH}$ KBs. In what follows, it will prove convenient to work with $\mathcal{ELH}$ TBoxes which are in *normal form*, which means that all concept inclusions are of one of the following forms:

$$A \sqsubseteq B \quad A_1 \sqcap A_2 \sqsubseteq B \quad A \sqsubseteq \exists r.B \quad \exists r.B \sqsubseteq A$$

with $A, A_1, A_2, B \in \mathsf{N_C} \cup \{\top\}$. This can be done without loss of generality since it is well-known (cf. [Baader *et al.*, 2005]) that for every $\mathcal{ELH}$ TBox $\mathcal{T}$, one can construct in polynomial time an $\mathcal{ELH}$ TBox $\mathcal{T}'$ in normal form (possibly using new concept names) which is a model conservative extension of $\mathcal{T}$, i.e. is such that $\mathcal{T}' \models \mathcal{T}$ and every model of $\mathcal{T}$ can be expanded to a model of $\mathcal{T}'$. *We assume henceforth that all $\mathcal{ELH}$ TBoxes are in normal form.*

We recall the definition of canonical models in $\mathcal{ELH}$ (see e.g. [Lutz *et al.*, 2009]). Given an $\mathcal{ELH}$ KB $(\mathcal{T},\mathcal{A})$, the domain $\Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ of the canonical model $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ consists those objects of the form $ar_1A_1\ldots r_nA_n$ ($n \geq 0$), such that $a \in \mathsf{Ind}(\mathcal{A})$, each A_i is a concept name, and each r_i is a role name, and the following conditions hold:

- if $n \geq 1$, then $\mathcal{T},\mathcal{A} \models \exists r_1.A_1(a)$;
- for $1 \leq i < n$, $\mathcal{T} \models A_i \sqsubseteq \exists r_{i+1}.A_{i+1}$.

If $w \in \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} \setminus \mathsf{Ind}(\mathcal{A})$, then we denote by $\mathsf{tail}(w)$ the final concept name in w, and define $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ by taking:

$$\begin{aligned}
a^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} &= a \text{ for all } a \in \mathsf{Ind}(\mathcal{A})\\
A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} &= \{a \in \mathsf{Ind}(\mathcal{A}) \mid \mathcal{T},\mathcal{A} \models A(a)\}\\
&\qquad \cup\ \{w \in \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} \setminus \mathsf{Ind}(\mathcal{A}) \mid \mathcal{T} \models \mathsf{tail}(w) \sqsubseteq A\}\\
r^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} &= \{(a,b) \mid s(a,b) \in \mathcal{A} \text{ for some } s \text{ with } \mathcal{T} \models s \sqsubseteq r\} \cup\\
&\qquad \{(w_1,w_2) \mid w_2 = w_1 s\, A \text{ and } \mathcal{T} \models s \sqsubseteq r\}
\end{aligned}$$

The following technical lemmas establish some properties of tree witnesses in $\mathcal{ELH}$ and will play an important role in proof of Proposition 12.

Lemma 8. *Let $(\mathcal{T},\mathcal{A})$ be a consistent $\mathcal{ELH}$ knowledge base, let $\alpha = r(x,y)$ be an atom in a CQ ρ, and let D_α and $\sim_\alpha$ be as defined in Section 3. If π is a match for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ with $\pi(y) \notin \mathsf{Ind}(\mathcal{A})$, then:*

1. *for every $u \in D_\alpha$, $\pi(x)$ is a prefix of $\pi(u)$*
2. *for every $u \in D_\alpha$ with $\pi(u) = \pi(x)$, $u \sim_\alpha x$*
3. *for every pair $u, u' \in D_\alpha$ with $u \sim_\alpha u'$, $\pi(u) = \pi(u')$*
4. *if $u' \in D_\alpha$ and $u' \not\sim_\alpha x$, then there is some atom $s(v,v') \in \rho$ such that $v, v' \in D_\alpha$ and $u' \sim_\alpha v'$*

Proof. Suppose that π is a match for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\pi(y) \notin \mathsf{Ind}(\mathcal{A})$. Fix a particular sequence of rule applications which generates the set D_α, and let $D_0 = \{x,y\}, D_1, \ldots, D_\ell = D_\alpha$ and $\sim_0, \sim_1, \ldots, \sim_\ell = \sim_\alpha$ be the corresponding sequences of sets of variables and equivalence relations (i.e. at stage i, we have the set D_i and equivalence relation $\sim_i$). More precisely, $\sim_i$ denotes the smallest equivalence relation over D_i which contains (u,u') whenever $u \sim u'$ has been asserted at stage $j \leq i$. We prove by induction that for all $1 \leq i \leq \ell$:

(i) if $u \in D_i$, then $\pi(u)$ has prefix $\pi(x)$

(ii) if $u \in D_i$ and $\pi(u) = \pi(x)$, then $u \sim_i x$

(iii) if $u \sim_i u'$, then $\pi(u) = \pi(u')$

(iv) if $u' \in D_i$ and $u' \not\sim_i x$, then there is some atom $s(v,v') \in \rho$ such that $v, v' \in D_i$ and $u' \sim_i v'$

The base case ($i = 0$) is straightforward: it follows from the structure of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ that $\pi(x)$ is a prefix of $\pi(y)$, (ii) trivially holds since $\pi(x) \neq \pi(y)$, (iii) trivially holds since $\sim_0$ contains only the singleton equivalence classes $\{x\}$ and $\{y\}$, and (iv) holds by taking the atom $r(x,y)$. Now suppose that properties (i)-(iv) hold for all $1 \leq i < k$, and let us show they continue to hold for $i = k$. First suppose that D_k was obtained from D_{k-1} by an application of the first rule. Then there must exist an atom $s(v,u) \in \rho$ such that $v \in D_{k-1}$, $v \not\sim_{k-1} x$, and $u \notin D_{k-1}$. By the induction hypothesis and the fact that $v \not\sim_{k-1} x$, $\pi(v)$ must have $\pi(x)$ as a proper prefix. It follows from the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ and the fact that $\pi(v)$ is in the anonymous part that $\pi(u) = \pi(v)rA$ for some r with $\mathcal{T} \models r \sqsubseteq s$, so $\pi(u)$ also has $\pi(x)$ as a proper prefix. This shows that D_k verifies properties (i) and (ii). For property (iii), we simply remark that the equivalence relation is not modified by the first rule, and so (iii) follows directly from the induction hypothesis. Property (iv) is witnessed by the atom $s(v,u)$.

The other possibility is that the second rule was applied. Then there must exist atoms $s(u,u'), t(z,z') \in \rho$ such that $u', z, z' \in D_{k-1}$, $u' \sim_{k-1} z'$, and $u \notin D_{k-1}$. The second rule will add u to D_k and the pair (u,z) to the equivalence relation. By the induction hypothesis, we know that $\pi(u') = \pi(z')$ and that $\pi(u'), \pi(z), \pi(z')$ all have prefix $\pi(x)$. Because of the tree structure of the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, the object $\pi(u)$ is the parent of $\pi(u')$, and likewise, $\pi(z)$ is the parent of $\pi(z')$. Since $\pi(u') = \pi(z')$ and each object has exactly one parent, we can infer that $\pi(u) = \pi(z)$, so properties (i) and (iii) hold for D_k. For (ii), note that if $\pi(u) = \pi(x)$, then $\pi(z) = \pi(x)$, so by the induction hypothesis, $z \sim_{k-1} x$. Since we also have $u \sim_k z$, we obtain $u \sim_k x$, as desired. Finally, for (iv), the induction hypothesis yields an atom $s'(v,v') \in \rho$ such that $v, v' \in D_{k-1}$ and $z \sim_{k-1} v'$. It follows that $u \sim_k v'$, so property (iv) holds also for $i = k + 1$. □

Lemma 9. *Let $(\mathcal{T},\mathcal{A})$ be a consistent $\mathcal{ELH}$ knowledge base, and let π be a match for a CQ ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. If $r(x,y) \in \rho$ and $\pi(y) \notin \mathsf{Ind}(\mathcal{A})$, then:*

- *the tree witness f for $r(x,y)$ exists*
- *for every z in domain of f, either $\pi(z) \in \mathsf{Ind}(\mathcal{A})$ and $\mathcal{T},\mathcal{A} \models \mathsf{conc}_f(f(z))(\pi(z))$, or $\pi(z) \notin \mathsf{Ind}(\mathcal{A})$ and $\mathcal{T} \models \mathsf{tail}(\pi(z)) \sqsubseteq \mathsf{conc}_f(f(z))$.*

Proof. Suppose that $\alpha = r(x,y) \in \rho$ and $\pi(y) \notin \mathsf{Ind}(\mathcal{A})$. Let D_α, $\sim_\alpha$, ρ_α, and G_α be as defined in Section 3. The tree witness for $r(x,y)$ exists just in the case that G_α is acyclic, so suppose for a contradiction that G_α contains a cycle. Then there exists a sequence $[z_1], \ldots, [z_n]$ of equivalence classes under $\sim_\alpha$ such that:

- for every $1 \leq i < n$, there exist $u_i \in [z_i]$ and $u'_i \in [z_{i+1}]$ such that ρ contains an atom $r_i(u_i, u'_i)$
- there exist $u_n \in [z_n]$ and $u'_n \in [z_1]$ such that ρ contains an atom $r_n(u_n, u'_n)$

By point 3 of Lemma 8, $u \sim_\alpha u'$ implies $\pi(u) = \pi(u')$. It follows that $\pi(u'_i) = \pi(u_{i+1})$ for $1 \leq i < n$, and $\pi(u'_n) = \pi(u_1)$, and so $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ contains the cycle $r_1(\pi(u_1),\pi(u_2)), r_2(\pi(u_1),\pi(u_2)), \ldots, r_n(\pi(u_n),\pi(u_1))$. This contradicts the fact that by point 1 of Lemma 8, the $\pi(u_i)$ all belong to subtree of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ rooted at $\pi(x)$, which is cycle-free.

Now that we have shown that the tree witness f for $r(x,y)$ exists, we prove the second point by induction on the codepth of elements in the range of f. For the base case, assume that $f(z)$ is a leaf. Then $\mathsf{conc}_f(f(z)) = \top$, and the statement trivially holds. For the induction step, suppose that

$f(z)$ has children $f(z_1), \dots, f(z_n)$ with $f(z_i) = f(z)M_iN_i$, and we have already established the statement for each of the $f(z_i)$. Consider some $1 \leq i \leq n$. From the definition of f, we know that $M_i = \{s \mid s(u, u') \in \rho, u \in [z], u' \in [z_i]\}$. By point 3 of Lemma 8 and the fact that π is a match for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, we know that for every $s \in M_i$, $(\pi(z), \pi(z_i)) \in s^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. As $N_i = \{A \mid A(u) \in \rho, u \in [z_i]\}$, we also know that $\pi(z_i) \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ for every $A \in N_i$. It follows from the construction of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ that $\pi(z_i) = \pi(z)tB$ for $t \in \mathsf{N_R}$ and $B \in \mathsf{N_C}$ satisfying:

- $\mathcal{T} \models \mathsf{tail}(\pi(z)) \sqsubseteq \exists t.B$ (or $\mathcal{T}, \mathcal{A} \models \exists t.B(\pi(z))$ if $\pi(z) \in \mathsf{Ind}(\mathcal{A})$)
- $\mathcal{T} \models t \sqsubseteq s$ for every $s \in M_i$
- $\mathcal{T} \models B \sqsubseteq A$ for every $A \in N_i$

By the induction hypothesis, we also have that $\mathcal{T} \models \mathsf{tail}(\pi(z_i)) \sqsubseteq \mathsf{conc}_f(f(z_i))$, hence $\mathcal{T} \models B \sqsubseteq \mathsf{conc}_f(f(z_i))$. We have thus shown that

$$\mathcal{T} \models \mathsf{tail}(\pi(z)) \sqsubseteq \exists(\bigsqcap_{r \in M_i} r).(\bigsqcap_{A \in N_i} A \sqcap \mathsf{conc}_f(f(z_i)))$$

if $\pi(z) \notin \mathsf{Ind}(\mathcal{A})$, and

$$\mathcal{T}, \mathcal{A} \models \exists(\bigsqcap_{r \in M_i} r).(\bigsqcap_{A \in N_i} A \sqcap \mathsf{conc}_f(f(z_i)))(\pi(z))$$

in the case that $\pi(z) \in \mathsf{Ind}(\mathcal{A})$. Since this holds for all $1 \leq i \leq n$, by the definition of $\mathsf{conc}_f(f(z))$, we obtain either $\mathcal{T} \models \mathsf{tail}(\pi(z)) \sqsubseteq \mathsf{conc}_f(f(z))$ or $\mathcal{T}, \mathcal{A} \models \mathsf{conc}_f(f(z))(\pi(z))$, depending on whether $\pi(z) \in \mathsf{Ind}(\mathcal{A})$. □

Lemma 10. *Let $(\mathcal{T}, \mathcal{A})$ be a consistent $\mathcal{ELH}$ knowledge base, let π be a match for a connected CQ ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, and let x, y be such that $\pi(x)$ is the proper prefix of $\pi(y)$. Then one can find tree witnesses $f_1, \dots, f_k$ of $r_1(z_1, y_1), \dots, r_k(z_k, y_k) \in \rho$ such that:*

- *$\pi(z_i) = \pi(x)$ for every $1 \leq i \leq k$*
- *if $\pi(u)$ has $\pi(x)$ as a proper prefix, then u belongs to the domain of exactly one f_i*

Proof. Suppose that $(\mathcal{T}, \mathcal{A})$, ρ, π, and x satisfy the hypotheses of the lemma. Initialize i to 1 and Ω to the set of atoms $r(z, y) \in \rho$ such that $\pi(z) = \pi(x)$ and $\pi(y) = \pi(z)r$ (by our assumptions, there must be at least one such atom). We perform the following procedure:

Step 1: Choose an element $r_i(z_i, y_i)$ from Ω, and let f_i be the tree witness for $r_i(z_i, y_i)$ in ρ.

Step 2: Remove from Ω all atoms $s(z, y)$ such that y is in the domain of f_i.

Step 3: If $\Omega \neq \emptyset$, then increment i and return to Step 1.

It follows from Lemma 9 and the definition of Ω together that the tree witnesses in Step 1 exist, so the above procedure is well-defined. Since the atom in Step 1 is removed from Ω in Step 2, we eventually reach $\Omega = \emptyset$, so the procedure always terminates. Our aim is to show that the procedure yields a sequence of tree witnesses $f_1, \dots, f_k$ of $r_1(z_1, y_1), \dots, r_k(z_k, y_k)$ in ρ which satisfy the conditions of the lemma. The first condition trivially follows from the definition of Ω. For the second condition, suppose for a contradiction that $\pi(u)$ has $\pi(x)$ as a proper prefix, yet u does not belong to the domain of any tree witness f_i. By connectedness of ρ, we can find a sequence of variables $v_0, v_1, \dots, v_\ell$ with $v_\ell = u$ such that:

- $\pi(v_0) = \pi(x)$
- for every $1 \leq i < \ell$, $\pi(v_i)$ has $\pi(x)$ as a proper prefix
- ρ contains an atom $\alpha = s_0(v_0, v_1)$
- for every $1 \leq i < \ell$, ρ contains an atom of the form $s_i(v_i, v_{i+1})$ or $s_i(v_{i+1}, v_i)$

The atom $\alpha = s_0(v_0, v_1)$ belongs to Ω at the start of the procedure, and so it must be removed from Ω during some iteration. The first possibility is that α is the selected atom at iteration j, and f_j is the tree witness for α. It follows from the last three items and the definition of D_α that D_α contains the variable $v_\ell = u$, and so u belongs to the domain of f_j. The other possibility is that α is removed at iteration j, but another atom α' was selected at that iteration. In this case, f_j is the tree witness for α', and the removal of α at stage j implies that v_0 and v_1 belong to the domain of f_j. Again, by using the definition of D_α, we find that $v_\ell = u$ belongs to the domain of f_j.

Next suppose for a contradiction that $\pi(u)$ has $\pi(x)$ as a proper prefix and belongs both to the domains of f_{j_1} and f_{j_2} ($j_1 < j_2$). We recall that f_{j_1} and f_{j_2} are tree witnesses of $\alpha_{j_1} = r_{j_1}(z_{j_1}, y_{j_1})$ and $\alpha_{j_2} = r_{j_2}(z_{j_2}, y_{j_2})$ respectively. Since $\pi(x)$ is a proper prefix of $\pi(u)$, we have $\pi(x) \neq \pi(u)$. By point 3 of Lemma 8, we must have $u \not\sim_{\alpha_{j_1}} z_{j_1}$ and $u \not\sim_{\alpha_{j_2}} z_{j_2}$. By the definition of tree witnesses, the presence of u in the domain of f_{j_2} implies that there exist sequences of variables $v_1, \dots, v_\ell$ such that:

- $v_1 = y_{j_2}$ and $v_\ell = u$
- for every $1 \leq i \leq \ell_m$, v_i is in the domain $D_{\alpha_{j_2}}$ of f_{j_2} and $v_i \not\sim_{\alpha_{j_2}} z_{j_2}$
- for every $1 \leq i < \ell_m$, ρ contains an atom of the form $s_i(v_i, v_{i+1})$ or $s_i(v_{i+1}, v_i)$

Since the atom $\alpha_{j_2} = r_{j_2}(z_{j_2}, y_{j_2})$ was still present in Ω at iteration $j_2 > j_1$, we know that $v_1 = y_{j_2} \notin D_{\alpha_1}$. Let p be such that $v_p \notin D_{\alpha_1}$ and $v_{p'} \in D_{\alpha_1}$ for every $p < p' \leq \ell$. By the third item above, either ρ contains an atom $s_p(v_p, v_{p+1})$ or an atom $s_p(v_{p+1}, v_p)$. First suppose that ρ contains $s_p(v_p, v_{p+1})$. If $v_{p+1} \sim_{\alpha_{j_1}} z_{j_1}$, then by point 3 of Lemma 8, $\pi(v_{p+1}) = \pi(z_{j_1}) = \pi(x)$. By point 2 of Lemma 8 and the fact that $v_{p+1} \in D_{\alpha_2}$, we must have $v_{p+1} \sim_{\alpha_{j_2}} z_{j_2}$, contradicting the second item above. If $v_{p+1} \not\sim_{\alpha_{j_1}} z_{j_1}$, then we can apply point 4 of Lemma 8 to get an atom $s'(w, w') \in \rho$ such that $w, w' \in D_{\alpha_1}$ and $w' \sim_{\alpha_{j_1}} v_{p+1}$. It follows that the second rule is applicable so v_p must appear in the domain of f_{j_1}, a contradiction. It must thus be the case that ρ contains an atom $s_p(v_{p+1}, v_p)$. As $v_{p+1} \in D_{\alpha_1}$ but $v_p \notin D_{\alpha_1}$, the first rule for D_{α_1} must not be applicable to the atom $s_p(v_{p+1}, v_p)$, which implies that $v_{p+1} \sim_{\alpha_{j_1}} z_{j_1}$. Using the same argument as above, we can show that $v_{p+1} \sim_{\alpha_{j_2}} z_{j_2}$, a contradiction. We can thus conclude that no $\pi(u)$ which contains $\pi(x)$ as a proper prefix can belong to more than one f_i. □

Lemma 11. *Let $(\mathcal{T}, \mathcal{A})$ be a consistent $\mathcal{ELH}$ knowledge base, ρ be a CQ, f be the tree witness for $\alpha = r(x, y)$ in ρ, and C be the set of concept names A such that $A([x]) \in \rho_\alpha$. Then for every $o \in \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$:*

$$o \in \left(\bigsqcap_{A\in C} A \sqcap \mathsf{conc}_f(\varepsilon)\right)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} \qquad \textit{iff} \qquad o \in \mathsf{ans}(\rho_\alpha, \mathcal{I}_{\mathcal{T},\mathcal{A}})$$

where we take $[x]$ as the unique answer variable in ρ_α.

Proof. Straightforward using induction and the definition of the concept $\mathsf{conc}_f(\varepsilon)$ and query ρ_α. □

We now proceed to the proof of Proposition 12.

Proposition 12. *Let ρ be a CQ, let $(\mathcal{T}, \mathcal{A})$ be a consistent $\mathcal{ELH}$ knowledge base, and let $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ be the interpretation defined in Section 3. Then for every tuple $\vec{a}$ of individuals, $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A}})$ iff $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A},\rho})$.*

Proof. For the first direction, suppose that $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A}})$. By Fact 1, there exists a match π for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\pi(\vec{x}) = \vec{a}$, with $\vec{x}$ the answer variables of ρ. Let ρ' be a maximally connected component of ρ. There are two possibilities:

- ρ' contains a variable v such that $\pi(v) \in \mathsf{Ind}(\mathcal{A})$
- all variables in ρ' are mapped by π to the anonymous part.

We consider first the case in which there is a variable v in ρ' with $\pi(v) \in \mathsf{Ind}(\mathcal{A})$. Let b be an individual such that there exists a variable u in ρ' and a non-empty word w with $\pi(u) = bw$. Let ρ_b be the restriction of ρ' to variables u such that $\pi(u)$ has prefix b. As ρ' is connected, ρ_b must also be connected. By Lemma 10, we can find tree witnesses $f_1, \ldots, f_k$ of $\alpha_1 = r_1(z_1, y_1), \ldots, \alpha_k = r_k(z_k, y_k) \in \rho'$ such that:

- $\pi(z_i) = b$ for every $1 \leq i \leq k$
- if $\pi(u)$ has b as a proper prefix, then u belongs to the domain of exactly one f_i

From Lemma 9 and the facts that $f_i(z_i) = \varepsilon$ and $\pi(z_i) = b$, we have that $\mathcal{T}, \mathcal{A} \models \mathsf{conc}_{f_i}(\varepsilon)(b)$ for every $1 \leq i \leq k$. It follows that for every $1 \leq i \leq k$, and every w in the range of f_i, the object $b\alpha_i w$ belongs to $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Define a mapping μ_b by setting $\mu(u) = b$ if $\pi(u) = b$, and otherwise setting $\mu_b(u) = b\alpha_i f_i(u)$, where i is such that the domain of f_i contains u. By the second item above, every variable u in ρ_b with $\pi(u) \neq b$ must belong to the domain of exactly one f_i, so μ_b is well-defined.

We show next that μ_b is a match for ρ_b in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Let $A(u)$ be a concept atom in ρ_b. If $\mu_b(u) = b$, then $\pi(u) = b$. Since π is a match for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, we must have $b \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$, hence $\mathcal{T}, \mathcal{A} \models A(b)$ by Fact 1. From the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$, we get $\mu_b(u) = b \in A^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$. Suppose instead that $\mu_b(u)$ takes the form $b\alpha_i wMN$. Then it follows from the definition of μ_b that $f_i(u) = wMN$. By the definition of tree witnesses, we must have $A \in N$, hence $b\alpha_i wMN \in A^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$. Next consider a role atom $r(u, u') \in \rho_b$, and let f_i be the (unique) tree witness containing both u and u'. Then $f_i(u')$ must take the form $f_i(u)MN$ for some M, N with $r \in M$. If $f_i(u) = \varepsilon$, then $u \sim_{\alpha_i} z_i$, so by point 3 of Lemma 8, we have $\pi(u) = \pi(z_i) = b$. It follows that $\mu_b(u) = b$ and $\mu_b(u') = b\alpha_i MN$. If $f_i(u) \neq \varepsilon$, then $\mu_b(u) = b\alpha_i f_i(u)$ and $\mu_b(u') = b\alpha_i f_i(u') = b\alpha_i f_i(u)MN$. In either case, the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ yields $(\mu_b(u), \mu_b(u')) \in r^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$. We have thus shown that μ_b is a match for ρ_b in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$.

Define a mapping μ from the variables of ρ' to $\Delta^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$ by letting $\mu(u) = \pi(u)$ if $\pi(u) \in \mathsf{Ind}(\mathcal{A})$, and otherwise setting $\mu(u) = \mu_b(u)$ where b is the unique individual such that $\pi(u) = bw$. It is not hard to show that μ is a match for ρ' in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Moreover, μ coincides with π on the answer variables of ρ which appear in ρ'.

Let us next consider the other possibility, which is that all variables in ρ' are mapped by π to the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. It follows from the structure of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ and the connectedness of ρ' that we can find some variable x such that $\pi(x)$ is a prefix of $\pi(u)$, for every variable u in ρ'. By Lemma 10, we can find tree witnesses $f_1, \ldots, f_k$ of $\alpha_1 = r_1(z_1, y_1), \ldots, \alpha_k = r_k(z_k, y_k) \in \rho'$ such that:

- $\pi(z_i) = \pi(x)$ for every $1 \leq i \leq k$
- if $\pi(u)$ has $\pi(x)$ as a proper prefix, then u belongs to the domain of exactly one f_i

Let B be the concept name $\mathsf{tail}(\pi(x))$. Using Lemma 9 and the fact that $f_i(z_i) = \varepsilon$, we have $\mathcal{T} \models B \sqsubseteq \mathsf{conc}_{f_i}(\varepsilon)$ for every $1 \leq i \leq k$. It follows that for every $1 \leq i \leq k$, and every w in the range of f_i, the object $B\alpha_i w$ belongs to $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Define a mapping μ by setting $\mu(u) = B$ if $\pi(u) = \pi(x)$ and otherwise setting $\mu(u) = B\alpha_i f_i(u)$, with i such that u is in the domain of f_i. Using arguments very similar to those above, we can show that μ is well-defined and defines a match for ρ' in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$.

We have thus shown that every maximally connected component ρ' of ρ has a match in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ such that any answer variables are sent to the corresponding individuals from $\vec{a}$. By combining these matches, we obtain a match for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ which sends $\vec{x}$ to $\vec{a}$, yielding $\vec{a} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A},\rho})$.

For the other direction, it is sufficient to exhibit a homomorphism from $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ to $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. We prove the existence of such an homomorphism by induction on the length of objects in the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$. Specifically, we define the length of an object $\sigma \in \mathsf{Ind}(\mathcal{A}) \cup \mathsf{N_C}$ as 0 and the length of $\sigma\alpha M_1 N_1 \ldots M_n N_n$ as n. We use $\Delta_i^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ to denote the set of objects in $\Delta^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$ of length at most i, and let m be the maximum length of any object in $\Delta^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$. Our inductive argument will show that for all $0 \leq i \leq m$, there is a homomorphism h from the restriction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ to objects in $\Delta_i^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ to $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ which satisfies the following conditions:

- if $o = a\alpha w \in \Delta_i^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ and f is the tree witness for α in ρ, then $h(o) \in \mathsf{conc}_f(w)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$
- if $o = B\alpha w \in \Delta_i^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ and f is the tree witness for α in ρ, then $h(o) \in \mathsf{conc}_f(w)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$

For the first base case, we let $h_0(a) = a$ for all $a \in \mathsf{Ind}(\mathcal{A})$, and for every $B \in \Delta^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$, we let $h_0(B)$ be any object d in

$\Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ with $\mathsf{tail}(d) = B$ (such an object must exist, otherwise B would not belong to $\Delta^{\mathcal{I}}_{\mathcal{T},\mathcal{A},\rho}$). It is easily verified that h_0 is a homomorphism from the restriction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ to $\Delta_0^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ to $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ and that it trivially satisfies the two conditions. For the second base case, we start by setting $h_1(o) = h_0(o)$ for all objects with length 0. Then consider an object of length 1 of the form $a\alpha MN$. The presence of $a\alpha MN$ in $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ tells us that the tree witness f for α in ρ contains MN and $\mathcal{T},\mathcal{A} \models \mathsf{conc}_f(\varepsilon)(a)$. Using the definition of $\mathsf{conc}_f(\varepsilon)$, we obtain

$$\mathcal{T},\mathcal{A} \models \left(\exists \bigsqcap_{r\in M} r.(\bigsqcap_{A\in N} A \sqcap \mathsf{conc}_f(MN))\right)(a)$$

Consequently, there must exist some $d \in \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ such that $(a,d) \in r^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ for every $r \in M$, $d \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ for every $A \in N$, and $d \in \mathsf{conc}_f(MN)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. We let $h_1(a\alpha MN) = d$. Next consider an object of length 1 of the form $B\alpha MN$. By the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$, the presence of $B\alpha MN$ implies that the tree witness f for α has MN in its range, and that $\mathsf{conc}_f(\varepsilon)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ contains some element with tail concept B. In particular, this means that $\mathcal{T} \models B \sqsubseteq \mathsf{conc}_f(\varepsilon)$, so $h_0(B) \in \mathsf{conc}_f(\varepsilon)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. Using the definition of $\mathsf{conc}_f(\varepsilon)$, we get

$$h_0(B) \in \left(\exists \bigsqcap_{r\in M} r.(\bigsqcap_{A\in N} A \sqcap \mathsf{conc}_f(MN))\right)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$$

Consequently, there must exist some $d' \in \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ such that $(h_0(B), d') \in r^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ for every $r \in M$, $d' \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ for every $A \in N$, and $d' \in \mathsf{conc}_f(MN)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. We let $h_1(B\alpha MN) = d'$. It is easily verified that h_1 is a homomorphism and that it satisfies the above two conditions.

For the induction step, we suppose that the statement holds for $1 \leq i = k < m$, that is, there exists a homomorphism h_k from the restriction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ to $\Delta_k^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ to $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ which satisfies the required conditions. We let $h_{k+1}(o) = h_k(o)$ for all objects $o \in \Delta_k^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$. For an object $o = \sigma\alpha wMN \in \Delta_{k+1}^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}} \setminus \in \Delta_k^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$, we let f be the tree witness for α in ρ. By our induction hypothesis, $h_k(\sigma\alpha w) \in \mathsf{conc}_f(w)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$, hence

$$h_k(\sigma\alpha w) \in \left(\exists \bigsqcap_{r\in M} r.(\bigsqcap_{A\in N} A \sqcap \mathsf{conc}_f(wMN))\right)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$$

It follows that there exists some $d \in \Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ such that $(h_k(\sigma\alpha w), d) \in r^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ for every $r \in M$, $d \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ for every $A \in N$, and $d \in \mathsf{conc}_f(wMN)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. We set $h_{k+1}(\sigma wMN) = d$. It is easy to see that the mapping h_{k+1} thus defined is a homomorphism from the restriction of $\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}$ to $\Delta_k^{\mathcal{I}_{\mathcal{T},\mathcal{A},\rho}}$ to $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. Moreover, the two conditions hold since we have defined h_{k+1} so that $h_{k+1}(\sigma\alpha wMN) \in \mathsf{conc}_f(wMN)^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. □

B Proof of Theorem 5

Throughout this section, we assume that ρ is a rooted a-acyclic CQ and $\mathcal{K} = (\mathcal{T},\mathcal{A})$ is a DL-Lite KB. We recall that $\rho|x$ is the query whose answer variables are $\{x\} \cup \vec{x}_\rho$ and whose body is obtained by restricting the body of ρ to the atoms whose arguments among x and its descendants. If y is a child of x, then the query $\rho|xy$ is defined by adding to $\rho|y$ all role atoms in ρ which contain both x and y, and taking $\{x\} \cup \vec{y}^{+}_{\rho}$ as answer variables.

Also recall that the datalog program $\mathsf{rew}_{\mathcal{T}}(\rho)$ consists of the set of rules P and the goal relation q.

Lemma 13. *For every concept name A appearing in ρ, $\mathsf{cert}(A(x),\mathcal{K}) = \mathsf{ans}((P,q_A),\mathcal{I}_{\mathcal{A}})$. Likewise, for concepts $\exists R$ such that R (or its inverse) occurs in ρ, $\mathsf{cert}(\exists R(x),\mathcal{K}) = \mathsf{ans}((P,q_{\exists R}),\mathcal{I}_{\mathcal{A}})$.*

Proof. Straightforward: the rules defining q_A (resp. $q_{\exists R}$) correspond to the standard rewriting of $A(x)$(resp. $\exists R(x)$) with respect to $\mathcal{T}$, cf. [Calvanese *et al.*, 2007]. □

Lemma 14. *For every variable x in ρ,*

$$\mathsf{cert}(\rho|x,\mathcal{K}) = \mathsf{ans}((P,q_x),\mathcal{I}_{\mathcal{A}})$$

and for every pair of variables x, y with y a child of x,

$$\mathsf{cert}(\rho|xy,\mathcal{K}) = \mathsf{ans}((P,q'_y),\mathcal{I}_{\mathcal{A}}).$$

Proof. Let $\{T_1,\ldots,T_n\}$ be the set of trees associated with the query ρ, as described in Section 4. We recall that each variable in ρ belongs to exactly one T_i. We can thus define the co-depth of a variable x as the co-depth of x in the unique tree T_i containing x.

The proof proceeds by induction on the co-depth of variables. Specifically, we show:

- *Base case*: The first statement holds whenever x has co-depth 0.
- *First induction step*: if the second statement holds whenever y has co-depth at most k, then the first statement holds for all x with co-depth at most $k+1$.
- *Second induction step*: if the first statement holds whenever x has co-depth at most k, then the second statement holds whenever y has co-depth at most k.

It is not hard to see that the base case and two induction steps together imply the lemma.

We begin by establishing the base case. Take a variable x of co-depth 0. Since x has no descendants,

$$\rho|x = \bigwedge_{A(x)\in\rho} A(x) \wedge \bigwedge_{r(x,x)\in\rho} r(x,x)$$

and the only rule in P with head q_x is:

$$q_x(x) \leftarrow \bigwedge_{A(x)\in\rho} q_A(x) \wedge \bigwedge_{r(x,x)\in\rho} r(x,x)$$

It then suffices to apply Lemma 13 to obtain $\mathsf{cert}(\rho|x,\mathcal{K}) = \mathsf{ans}((P,q_x),\mathcal{I}_{\mathcal{A}})$.

For the first induction step, suppose that the second statement holds whenever y has co-depth at most k. Let x be a variable with co-depth $k+1$, and let $Y = \{y_1, \ldots, y_n\}$ be the children of x. By definition,

$$\rho|x = \bigwedge_{A(x)\in\rho} A(x) \wedge \bigwedge_{r(x,x)\in\rho} r(x,x) \wedge \bigwedge_{y\in Y} \rho|xy$$

and the (unique) rule in P defining the predicate q_x is as follows:

$$q_x(x,\vec{x}_\rho) \leftarrow \bigwedge_{A(x)\in\rho} q_A(x) \wedge \bigwedge_{r(x,x)\in\rho} r(x,x) \wedge \bigwedge_{y\in Y} q'_y(x,\vec{y}^{+}_\rho)$$

Applying the induction hypothesis to the variables in Y, we can infer that for every variable $y \in Y$,

$$\mathsf{cert}(\rho|xy,\mathcal{K}) = \mathsf{ans}((P,q'_y),\mathcal{I}_\mathcal{A}).$$

Putting this together with Lemma 13, we can conclude $\mathsf{cert}(\rho|x,\mathcal{K}) = \mathsf{ans}((P,q_x),\mathcal{I}_\mathcal{A})$.

For the second induction step, suppose that the first statement holds for variables having co-depth at most k. Let x,y be a pair of variables in ρ such that y is a child of x and has co-depth k. The ruleset P will contain the rule

$$\zeta_1: \quad q'_y(x,\vec{y}^{+}_\rho) \leftarrow \bigwedge_{r(x,y)\in\rho} r(x,y) \bigwedge_{s(y,x)\in\rho} s(y,x) \wedge q_y(y,\vec{y}_\rho)$$

and may additionally contain a rule

$$\zeta_2: \quad q'_y(x,\vec{u}) \leftarrow q_{\exists R}(x) \wedge \bigwedge_{z\in Z} q_z(x,\vec{z}_\rho)$$

if the following conditions are satisfied:

(i) there is an atom $R(x,y) \in \rho$ or $R^-(y,x) \in \rho$ and the tree witness $f_{R(x,y)}$ exists and is valid

(ii) for every u in domain of $f_{R(x,y)}$ with $f_{R(x,y)}(u) = wS$ and $A(u) \in \rho$, we have $\mathcal{T} \models \exists S^- \sqsubseteq A$

(iii) the set $Z = \{z \mid f_{R(x,y)}(z) = \varepsilon \wedge z \neq x\}$ contains all answer variables in the domain of $f_{R(x,y)}$

and $\vec{u}$ is obtained from $\vec{y}_\rho$ by replacing each $z \in Z$ by x.

For the first direction, suppose that $(a,\vec{c}) \in \mathsf{cert}(\rho|xy,\mathcal{K})$. By Fact 1, $\vec{t} \in \mathsf{ans}(\rho|xy,\mathcal{I}_{\mathcal{T},\mathcal{A}})$, and so there is a match π for $\rho|xy$ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\vec{t} = (\pi(x),\pi(\vec{y}^{+}_\rho))$ Using the fact that

$$\rho|xy = \bigwedge_{r(x,y)\in\rho} r(x,y) \bigwedge_{s(y,x)\in\rho} s(y,x) \wedge \rho|y$$

we obtain

$$(\pi(x),\pi(y)) \in \mathsf{ans}(\bigwedge_{r(x,y)\in\rho} r(x,y) \bigwedge_{s(y,x)\in\rho} s(y,x),\mathcal{I}_{\mathcal{T},\mathcal{A}})$$

and

$$(\pi(y),\pi(\vec{y}_\rho)) \in \mathsf{ans}(\rho|y,\mathcal{I}_{\mathcal{T},\mathcal{A}}).$$

Since x is an answer variable, we know that $\pi(x)$ is an ABox individual. However, $\pi(y)$ may either be an individual or an element of the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. We consider first the case in which $\pi(y)$ is an ABox individual. Since $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ and $\mathcal{I}_\mathcal{A}$ contain precisely the same role assertions involving individuals, we have

$$(\pi(x),\pi(y)) \in \mathsf{ans}(\bigwedge_{r(x,y)\in\rho} r(x,y) \bigwedge_{s(y,x)\in\rho} s(y,x),\mathcal{I}_\mathcal{A})$$

It follows that the restriction of π to the variables in $\{x,y\} \cup \vec{y}_\rho$ constitutes a match for the rule ζ_1. Next remark that since $(\pi(y),\pi(\vec{y}_\rho))$ is a tuple of individuals, we can apply Fact 1 to get $(\pi(y),\pi(\vec{y}_\rho)) \in \mathsf{cert}(\rho|y,\mathcal{K})$. The variable y has co-depth k, so the induction hypothesis applies and yields

$$(\pi(y),\pi(\vec{y}_\rho)) \in \mathsf{ans}((P,q_y),\mathcal{I}_\mathcal{A})$$

We have thus shown that the rule ζ_1 and the restriction of π to $\{x,y\} \cup \vec{y}_\rho$ satisfy all the required conditions, and so $(\pi(x),\pi(\vec{y}_\rho)) = \vec{t} \in \mathsf{ans}((P,q'_y),\mathcal{I}_\mathcal{A})$.

Next we consider the case in which $\pi(y)$ is an element in the anonymous part of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. It follows from the construction of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ and the fact that $(\pi(x),\pi(y)) \in R^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$ that $\pi(y) = \pi(x)R$ for some role R such that either $R(x,y) \in \rho$ or $R^-(y,x) \in \rho$. In particular, this means that the tree witness $f_{R(x,y)}$ exists and is valid, and moreover, $\pi(z) = \pi(x)f_{R(x,y)}(z)$ for each variable z in the domain of $f_{R(x,y)}$. If z is such that $f_{R(x,y)}(z) = wS$, then for every atom $A(z) \in \rho$, we have $\pi(z) \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$, hence $\mathcal{T} \models \exists S^- \sqsubseteq A$. Finally, we remark that all answer variables must be mapped to ABox individuals, so any answer variable z in the domain of $f_{R(x,y)}$ must be such that $f_{R(x,y)}(z) = \varepsilon$. Thus, conditions (i)-(iii) are satisfied, and so the rule $q'_y(x,\vec{u}) \leftarrow q_{\exists R}(x) \wedge \bigwedge_{z\in Z} q_z(x,\vec{z}_\rho)$ belongs to P. We then note that the presence of $\pi(x)R$ in the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ implies that $\pi(x) \in \exists R^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$, and hence that $\pi(x) \in \mathsf{cert}(\exists R(x),\mathcal{K})$. By Lemma 13,

$$\pi(x) \in \mathsf{ans}((P,q_{\exists R}),\mathcal{I}_\mathcal{A}).$$

We next remark that for each variable $z \in Z$, we must have $(\pi(z),\pi(\vec{z}_\rho)) \in \mathsf{ans}(\rho|z,\mathcal{I}_{\mathcal{T},\mathcal{A}})$. Using Fact 1 and the fact that $\pi(z) = \pi(x)$, we obtain $(\pi(x),\pi(\vec{z}_\rho)) \in \mathsf{cert}(\rho|z)$. Then since z has co-depth at most k, we can apply the induction hypothesis to get $\mathsf{cert}(\rho|z,\mathcal{K}) = \mathsf{ans}((P,q_z),\mathcal{I}_\mathcal{A})$, from which we can infer that

$$(\pi(x),\pi(\vec{z}_\rho)) \in \mathsf{ans}((P,q_z),\mathcal{I}_\mathcal{A})$$

and hence that

$$(\pi(x),\pi(\vec{u})) \in \mathsf{ans}((P,q'_y),\mathcal{I}_\mathcal{A}).$$

It then suffices to note that $\pi(\vec{u}) = \pi(\vec{y}^{+}_\rho)$, and hence $(\pi(x),\pi(\vec{u})) = \vec{t}$.

To complete the proof, we must show the other inclusion: $\mathsf{ans}((P,q'_y),\mathcal{I}_\mathcal{A}) \subseteq \mathsf{cert}(\rho|xy,\mathcal{K})$. Take some $\vec{t} \in \mathsf{ans}((P,q'_y),\mathcal{I}_\mathcal{A})$. The first possibility is that there is a match π for the rule ζ_1 in $\mathcal{I}_\mathcal{A}$ such that $(\pi(x),\pi(\vec{y}^{+}_\rho)) = \vec{t}$ and $(\pi(y),\pi(\vec{y}_\rho)) \in \mathsf{ans}((P,q_y),\mathcal{I}_\mathcal{A})$. As π is a match for ζ_1 in $\mathcal{I}_\mathcal{A}$, we must have

$$(\pi(x),\pi(y)) \in \mathsf{ans}(\bigwedge_{r(x,y)\in\rho} r(x,y) \bigwedge_{s(y,x)\in\rho} s(y,x),\mathcal{I}_\mathcal{A}).$$

Since $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ contains all role assertions from $\mathcal{I}_{\mathcal{A}}$, we also have

$$(\pi(x), \pi(y)) \in \mathsf{ans}(\bigwedge_{r(x,y)\in\rho} r(x,y) \bigwedge_{s(y,x)\in\rho} s(y,x), \mathcal{I}_{\mathcal{T},\mathcal{A}}).$$

We next remark that y has co-depth at most k, so we can apply the induction hypothesis and Fact 1 to obtain $(\pi(y), \pi(\vec{y}_\rho)) \in \mathsf{ans}(\rho|y, \mathcal{I}_{\mathcal{T},\mathcal{A}})$. It follows that π defines a match for $\rho|xy$ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. Another application of Fact 1 yields $\vec{t} = (\pi(y), \pi(\vec{y}_\rho)) \in \mathsf{cert}(\rho|xy, \mathcal{K})$.

The second possibility is that the rule ζ_2 belongs to P, and there is an assignment π to the variables in ζ_2 such that $\pi(x) \in \mathsf{ans}((P, q_{\exists R}), \mathcal{I}_{\mathcal{A}})$ and for each $z \in Z$, we have $(\pi(x), \pi(\vec{z}_\rho)) \in \mathsf{ans}((P, q_z), \mathcal{I}_{\mathcal{A}})$. Note that the presence of ζ_2 implies that conditions (i)-(iii) hold, and in particular, the tree witness $f_R(x, y)$ exists and is valid. Using Lemma 13, we obtain

$$\pi(x) \in \mathsf{cert}(\exists R(x), \mathcal{K}).$$

Then since each $z \in Z$ has co-depth at most k, we can apply the induction hypothesis to obtain

$$(\pi(x), \pi(\vec{z}_\rho)) \in \mathsf{cert}(\rho|z, \mathcal{K}).$$

It follows from Fact 1 and the definition of certain answers that for each $z \in Z$, we can find a match π_z for $\rho|z$ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\pi(u) = \pi_z(u)$ for all variables $u \in \{x\} \cup \vec{z}_\rho$. Define a new assignment π' as follows:

- $\pi'(u) = \pi(x) f_{R(x,y)}(u)$ if u is in the domain of $f_{R(x,y)}$
- $\pi'(u) = \pi_z(u)$ if u is a descendant of z in ρ

To see that π' is well-defined, note that a variable u can be the descendant of at most one z, and if u is the descendant of z, then it cannot belong to the domain of $f_{R(x,y)}$. Also note that every variable which appears in $\rho|xy$ is covered by one of the two items, and if $\pi'(u) = \pi(x) f_{R(x,y)}(u)$, then the validity of the tree witness $f_{R(x,y)}$ ensures that the object $\pi(x) f_{R(x,y)}(u)$ belongs to the domain of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. It remains to show that π' is a match for $\rho|xy$ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. Since $\pi' = \pi_z$ for all variables in $\rho|z$, we know that π' satisfies all atoms in $\cup_{z\in Z} \rho|z$. It thus remains to show that the atoms in $\rho|xy$ which are not in $\cup_{z\in Z} \rho|z$ are also satisfied. If $A(u) \in \rho|xy \setminus \cup_{z\in Z} \rho|z$, then we must have $f_{R(x,y)}(u) = wS$. Condition (ii) yields $\mathcal{T} \models \exists S^- \sqsubseteq A$, and the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ gives us $\pi'(u) = \pi(x)wS \in A^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. If instead we have a role atom $s(u, u') \in \rho|xy \setminus \cup_{z\in Z} \rho|z$, then u and u' both belong to the domain of $f_{R(x,y)}$. Moreover, from the definition of tree witnesses, we know that either (a) $f_{R(x,y)}(u') = f_{R(x,y)}(u)s$ or $f_{R(x,y)}(u) = f_{R(x,y)}(u')s^-$. In both cases, it follows from the definition of $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ that $(\pi'(u), \pi'(u')) \in s^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. We have thus shown that all atoms in $\rho|xy$ are satisfied by π', which means π' is a match for $\rho|xy$ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$. As $(\pi'(y), \pi'(\vec{y}_\rho)) = (\pi(y), \pi(\vec{y}_\rho))$ and $\vec{t} = (\pi(y), \pi(\vec{y}_\rho))$, we obtain the desired $\vec{t} \in \mathsf{cert}(\rho|xy, \mathcal{K})$. □

Theorem 5. $\mathsf{cert}(\rho, \mathcal{K}) = \mathsf{ans}(\mathsf{rew}_{\mathcal{T}}(\rho), \mathcal{I}_{\mathcal{A}})$.

Proof. We recall that $\mathsf{rew}_{\mathcal{T}}(\rho) = (P, q)$ and the only rule in P with head relation q is:

$$\theta : \quad q(\vec{x}) \leftarrow \bigwedge_{x\in X} q_x(x, \vec{x}_\rho) \wedge \bigwedge_{x_i, x_j \in \vec{x},\, r(x_i,x_j)\in\rho} r(x_i, x_j)$$

For the first direction, suppose that $\vec{t} \in \mathsf{cert}(\rho, \mathcal{K})$. By Fact 1, $\vec{t} \in \mathsf{ans}(\rho, \mathcal{I}_{\mathcal{T},\mathcal{A}})$, and so we can find a match π for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\pi(\vec{x}) = \vec{t}$. Note that π is a match also for the rule θ since all DL-atoms in θ are atoms in ρ. Further note that for every $x \in X$, the subquery $\rho|x$ is satisfied under π, so $(\pi(x), \pi(\vec{x}_\rho)) \in \mathsf{ans}(\rho|x, \mathcal{I}_{\mathcal{T},\mathcal{A}})$. Using Fact 1 and Lemma 14, we obtain $(\pi(x), \pi(\vec{x}_\rho)) \in \mathsf{ans}((P, q_x), \mathcal{I}_{\mathcal{A}})$. We have thus exhibited a match π for θ satisfying the required conditions, so we may conclude that $\vec{t} = \pi(\vec{x}) \in \mathsf{ans}(\mathsf{rew}_{\mathcal{T}}(\rho), \mathcal{I}_{\mathcal{A}})$.

For the second direction, suppose $\vec{t} \in \mathsf{ans}(\mathsf{rew}_{\mathcal{T}}(\rho), \mathcal{I}_{\mathcal{A}})$. Then it must be the case that there is a match π for rule θ in $\mathcal{I}_{\mathcal{A}}$ such that $\pi(\vec{x}) = \vec{t}$ and for every $x \in X$, $(\pi(x), \pi(\vec{x}_\rho)) \in \mathsf{ans}((P, q_x), \mathcal{I}_{\mathcal{A}})$. By Lemma 14, this means that for every variable $x \in X$, we have $(\pi(x), \pi(\vec{x}_\rho)) \in \mathsf{cert}(\rho|x, \mathcal{K})$. We can thus find for each $x \in X$, a match μ_x for $\rho|x$ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\mu_x(x) = \pi(x)$ and $\mu_x(\vec{x}_\rho) = \pi(\vec{x}_\rho)$. Define an assignment μ for the variables in ρ by setting $\mu(u) = \mu_x(u)$, where x is the root of the unique T_i containing u. By definition, μ satisfies all atoms in each of the queries ρ_x. All other atoms in ρ must take the form $s(u, u')$, with u and u' answer variables. Since π is a match for θ in $\mathcal{I}_{\mathcal{A}}$, we must have $(\pi(u), \pi(u')) \in s^{\mathcal{I}_{\mathcal{A}}}$, and hence $(\pi(u), \pi(u')) \in s^{\mathcal{I}_{\mathcal{T},\mathcal{A}}}$. As π and μ coincide on answer variables, we also have $(\mu(u), \pi(u')) \in s^{\mathcal{I}_{\mathcal{A}}}$, so $s(u, u')$ is satisfied by assignment μ. We have thus found a match μ for ρ in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ such that $\mu(\vec{x}) = \pi(\vec{x}) = \vec{t}$. Applying Fact 1, we obtain $\vec{t} \in \mathsf{cert}(\rho, \mathcal{K})$. □