Actively Learning $\mathcal{ELI}$ Queries under DL-Lite Ontologies

Maurice Funk[1], Jean Christoph Jung[2], and Carsten Lutz[1]

[1] Department of Computer Science, University of Bremen, Germany
[2] Department of Computer Science, University of Hildesheim, Germany

Abstract. We show that $\mathcal{ELI}$ queries (ELIQs) are learnable in polynomial time in the presence of a DL-Lite ontology $\mathcal{O}$, in Angluin's framework of active learning. When initially provided with a conjunctive query (CQ) that implies the target ELIQ under $\mathcal{O}$ (in the sense of query containment), it suffices for the learner to only pose membership queries to the oracle, but no equivalence queries. The initial CQ can be obtained by a single equivalence query and is available 'for free' in case that $\mathcal{O}$ does not pose any disjointness constraints on concepts. Our main technical result is that every $\mathcal{ELI}$ concept has only polynomially many most specific subsumers w.r.t. a DL-Lite ontology, generalizing a recent result about homomorphism frontiers by ten Cate and Dalmau.

1 Introduction

Constructing description logic (DL) concepts, ontologies, and queries can be challenging and costly, especially when logic expertise and domain knowledge are not in the same hands. This has prompted many approaches to *learning* such objects, including PAC learning [12,13,14], the construction of the least common subsumer (LCS) and the most specific concept (MSC) [4,6,7,19,28], and learning from data examples [16,18,22,23,27]. In recent years, there has been significant interest in applying *Angluin's framework of exact learning* in a DL context where a learner interacts in a game-like fashion with an oracle [1,2]. In particular, the learner may be a DL expert and the oracle a collaborating domain expert. The main aim is then to find an algorithm that enables the learner to construct the target object in polynomial time based on queries that it poses to the oracle, even when the oracle is not able to answer the queries in the most informative way.

The interest in exact learning in DLs started with an investigation of ontology learning in (the conference version of) [21], see also [20,26] and the survey [25]. This was recently complemented by studies of exactly learning DL concepts and queries: active learning of $\mathcal{ELI}$ concept queries (ELIQs) without ontologies is considered in [11] while [15] studies active learning of $\mathcal{EL}$ concept queries (ELQs), ELIQs, and restricted forms of conjunctive queries (CQs) in the presence of $\mathcal{EL}$

and $\mathcal{ELI}$ ontologies. The purpose of the current paper is to initiate the study of actively learning concepts and queries under ontologies formulated in DL-Lite, a prominent family of DLs that is featured in the OWL 2 family of ontology languages [3]. Our main result is that ELIQs, which can be viewed both as queries and as concepts to be used in an ontology, can be learned under DL-Lite ontologies in polynomial time even when the oracle can pose only a very basic kind of query to the oracle. To make this precise, we introduce the exact learning framework in more detail.

Learner and oracle both know and agree on the ontology $\mathcal{O}$ and the concept and role names that are available for constructing the target ELIQ q_T which must be satisfiable w.r.t. $\mathcal{O}$; we assume that this includes all concept and role names in $\mathcal{O}$. In a *membership query*, the learner provides an ABox $\mathcal{A}$ and a candidate answer $\bar{a}$ and asks whether $\mathcal{A}, \mathcal{O} \models q_T(\bar{a})$; the oracle faithfully answers "yes" or "no". In an *equivalence query*, the learner provides a hypothesis ELIQ q_H and asks whether q_H is equivalent to q_T under $\mathcal{O}$; the oracle answers "yes" or provides a counterexample, that is, an ABox $\mathcal{A}$ and tuple $\bar{a}$ such that $\mathcal{A}, \mathcal{O} \models q_T(\bar{a})$ and $\mathcal{A}, \mathcal{O} \not\models q_H(\bar{a})$ (*positive counterexample*) or vice versa (*negative counterexample*). One is then interested in *polynomial time learnability*, that is, whether there is a learning algorithm that constructs $q_T(\bar{x})$, up to equivalence w.r.t. $\mathcal{O}$, such that at any given time, the running time of the algorithm is bounded by a polynomial in the sizes of q_T, of $\mathcal{O}$, and of the largest counterexample given by the oracle so far. A weaker requirement is *polynomial query learnability* where only the sum of the sizes of the queries posed to the oracle up to the current time point has to be bounded by such a polynomial.

We can now state our main result more precisely. With *DL-Lite*, we generally refer to the basic member of the DL-Lite family [10] that admits inclusions between basic concepts, concept disjointness constraints, and role disjointness constraints; *DL-Lite*$^-$ then means the fragment without concept disjointness. Our main result is that ELIQs are polynomial time learnable using only membership queries under *DL-Lite*$^-$ ontologies, and that the same is true for *DL-Lite* ontologies provided that we have available an initial CQ q_H^0 such that $q_H^0 \subseteq_{\mathcal{O}} q_T$, that is, the answers to q_H^0 w.r.t. $\mathcal{O}$ are a subset of those to q_T w.r.t. $\mathcal{O}$ on every ABox $\mathcal{A}$. Such a q_H^0 can be obtained by a single initial equivalence query. We also observe that polynomial learnability using only membership queries fails in the presence of concept disjointness.

Let us mention two interesting perspectives on our results. First, they generalize the results in [11] about polynomial time learnability of ELIQs to the case with *DL-Lite* ontologies, in fact borrowing and extending crucial techniques from [11]. And second, the results in [15] demonstrate that inverse roles pose a significant challenge to polynomial time learnability. More precisely, [15] brings forward a polynomial time learning algorithm for symmetry-free ELIQs under $\mathcal{EL}$ ontologies where symmetry-free means that there is no subconcept of the form $\exists r.(C \sqcap \exists r^-.D)$ with r a role name. It is not clear at all how to generalize that algorithm to unrestricted ELIQs. Moreover, it is proved in [15] that ELQs are not polynomial query learnable under $\mathcal{ELI}$ ontologies. Thus, inverse roles tend to be challenging both in the query and in the ontology. In contrast, the result in this paper need not impose any restriction on the use of inverse roles. It seems relevant to recall here that *DL-Lite* is a fragment of $\mathcal{ELI}$.

A core technical result underlying our approach is that the *frontier* of an ELIQ q w.r.t. a *DL-Lite*$^-$ ontology is only of polynomial size and can be computed in polynomial time, generalizing a similar result from [11] that does not encompass ontologies. More precisely, a frontier of an ELIQ q w.r.t. a *DL-Lite* ontology $\mathcal{O}$ is a set of ELIQs $\mathcal{F}$ such that $q \subseteq_\mathcal{O} q_F$ and $q_F \not\subseteq_\mathcal{O} q$ for all $q_F \in \mathcal{F}$ and for all ELIQs q' with $q \subseteq_\mathcal{O} q'$ and $q' \not\subseteq_\mathcal{O} q$, there is a $q_F \in \mathcal{F}$ such that $q_F \subseteq_\mathcal{O} q'$. Note that if one thinks of q as an $\mathcal{ELI}$ concept, then $\mathcal{F}$ is the set of most specific subsumers of q w.r.t. $\mathcal{O}$.[1] Apart from being essential for our learning algorithm, there is another reason for why one may be interested in the frontier. In fact, it is observed in [11] that if an ELIQ q has a frontier of polynomial size, then q can be characterized up to equivalence by polynomially many data examples. Such an example takes the form $(\mathcal{A}, a)$ and is a positive example if $\mathcal{A} \models q(a)$ and a negative example otherwise. The same is true in the presence of ontologies.

Proof details are given in the appendix of the long version of this paper, available at `http://www.informatik.uni-bremen.de/tdki/research/papers.html`.

2 Preliminaries

Ontologies and ABoxes. Let $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_I}$ be countably infinite sets of *concept*, *role* and *individual names*. A *role* R is a role name r or the inverse r^- of a role name. A *basic concept* B is $\top$, a concept name A, or of the form $\exists R$, R a role. A *DL-Lite* ontology $\mathcal{O}$ is a finite set of (basic) *concept inclusions* $B_1 \sqsubseteq B_2$, *concept disjointness constraints* $B_1 \sqcap B_2 \sqsubseteq \bot$, and *role disjointness constraints* $R_1 \sqcap R_2 \sqsubseteq \bot$. A *DL-Lite*$^-$ *ontology* is a *DL-Lite* ontology that contains no concept disjointness constraints. A *DL-Lite* ontology is in *normal form* if all concept inclusions in it are of the form $A \sqsubseteq B$ or $B \sqsubseteq A$ with A a concept name or $\top$ and B a basic concept. An ABox $\mathcal{A}$ is a finite set of concept assertions $A(a)$ and role assertions $r(a, b)$ with A a concept name or $\top$, r a role name, and a, b individual names. We use $\mathsf{ind}(\mathcal{A})$ to denote the set of individual names used in $\mathcal{A}$.

The semantics is defined as usual in terms of *interpretations* $\mathcal{I}$, which we define to be a (possibly infinite and) non-empty set of concept and role assertions. We use $\Delta^\mathcal{I}$ to denote the set of individual names in $\mathcal{I}$, define $A^\mathcal{I} = \{a \mid A(a) \in \mathcal{I}\}$ for all $A \in \mathsf{N_C}$, and $r^\mathcal{I} = \{(a, b) \mid r(a, b) \in \mathcal{I}\}$ and $(r^-)^\mathcal{I} = \{(b, a) \mid r(a, b) \in \mathcal{I}\}$ for all $r \in \mathsf{N_R}$. We further set $\top^\mathcal{I} = \Delta^\mathcal{I}$ and $(\exists R)^\mathcal{I} = \{a \in \Delta^\mathcal{I} \mid \exists (a, b) \in R^\mathcal{I}\}$ for all roles R. This definition of interpretation is slightly different from the usual one, but equivalent;[2] its virtue is uniformity as every ABox is a finite interpretation. An interpretation $\mathcal{I}$ *satisfies* a concept inclusion $B_1 \sqsubseteq B_2$ if $B_1^\mathcal{I} \subseteq B_2^\mathcal{I}$, a concept

[1] One could equivalently say that $\mathcal{F}$ is the set of LCSs of a single concept which strictly generalize that concept.

[2] This depends on admitting assertions $\top(a)$ in ABoxes.

disjointness constraint $B_1 \sqcap B_2 \sqsubseteq \bot$ if $B_1^{\mathcal{I}} \cap B_2^{\mathcal{I}} = \emptyset$, and a role disjointness constraint $R_1 \sqcap R_2 \sqsubseteq \bot$ if $R_1^{\mathcal{I}} \cap R_2^{\mathcal{I}} = \emptyset$.

An interpretation is a *model* of a *DL-Lite* ontology or an ABox if it satisfies all concept inclusions, disjointness constraints and assertions in it. We write $\mathcal{O} \models B_1 \sqsubseteq B_2$ if every model of $\mathcal{O}$ satisfies the basic concept inclusion $B_1 \sqsubseteq B_2$ and $\mathcal{A}, \mathcal{O} \models B(a)$ if every model of $\mathcal{A}$ and $\mathcal{O}$ satisfies the concept assertion $B(a)$. An ABox $\mathcal{A}$ is *satisfiable* w.r.t. a *DL-Lite* ontology $\mathcal{O}$ if $\mathcal{A}$ and $\mathcal{O}$ have a common model.

A *signature* is a set of concept and role names, uniformly referred to as symbols. For any syntactic object O such as an ontology or an ABox, we use $\mathsf{sig}(O)$ to denote the symbols used in O and $||O||$ to denote the *size* of O, that is, the length of a representation of O as a word in a suitable alphabet.

Conjunctive Queries, ELIQs, Homomorphisms. A *conjunctive query* (CQ) takes the form $q(\bar{x}) \leftarrow \phi(\bar{x}, \bar{y})$ where ϕ is a conjunction of *concept atoms* $A(x)$, with $A \in \mathsf{N_C}$, and *role atoms* $r(x, y)$, with $r \in \mathsf{N_R}$ over variables $x, y \in \bar{x} \cup \bar{y}$. We may write $r^-(x, y)$ in place of $r(y, x)$. We refer to the variables in $\bar{x}$ as *answer variables* and to the variables in $\bar{y}$ as *quantified variables*. We use $\mathsf{var}(q)$ to denote the set of all variables in $\bar{x}$ and $\bar{y}$. We may view a CQ $q(\bar{x})$ as a set of atoms when convenient and write $r(x, y) \in q(\bar{x})$ to mean that $r(x, y)$ occurs in the conjunction ϕ.

A conjunctive query $q(\bar{x})$ is *unary* if $q(\bar{x})$ only has a single answer variable. A *cycle* in a CQ q is a sequence $R_1(x_1, x_2), \ldots, R_n(x_n, x_1)$ of distinct role atoms in q such that $x_1, \ldots x_n$ are distinct. An *ELIQ* is a unary CQ q that does not contain a cycle and such that the undirected graph $G_q = (\mathsf{var}(q), \{\{y, z\} \mid r(y, z) \in q\})$ is connected. Note that every ELIQ can be seen as an $\mathcal{ELI}$ concept in a straightforward way and vice versa; see [5] for a definition of $\mathcal{ELI}$ concepts. We use $\mathcal{A}_q$ to denote the ABox obtained from q by viewing variables as individuals and atoms as assertions. A CQ q is *satisfiable* w.r.t. a *DL-Lite* ontology $\mathcal{O}$ if $\mathcal{A}_q$ is satisfiable w.r.t. $\mathcal{O}$.

A *homomorphism* h from interpretation $\mathcal{I}_1$ to interpretation $\mathcal{I}_2$ is a mapping from $\Delta^{\mathcal{I}_1}$ to $\Delta^{\mathcal{I}_2}$ such that $d \in A^{\mathcal{I}_1}$ implies $h(d) \in A^{\mathcal{I}_2}$ and $(d, e) \in r^{\mathcal{I}_1}$ implies $(h(d), h(e)) \in r^{\mathcal{I}_2}$. We use $\mathsf{img}(h)$ to denote the set $\{e \in \Delta^{\mathcal{I}_2} \mid \exists d \in \Delta^{\mathcal{I}_1} \colon h(d) = e\}$. For $d_i \in \Delta^{\mathcal{I}_i}$, $i \in \{1, 2\}$, we write $\mathcal{I}_1, d_1 \to \mathcal{I}_2, d_2$ if there is a homomorphism h from $\mathcal{I}_1$ to $\mathcal{I}_2$ with $h(d_1) = d_2$. With a homomorphism from a CQ q to an interpretation $\mathcal{I}$, we mean a homomorphism from $\mathcal{A}_q$ to $\mathcal{I}$. For a unary CQ $q(x)$, we write $q(x) \to (\mathcal{I}, d)$ if there is a homomorphism h from q to $\mathcal{I}$ with $h(x) = d$. Let $q(x)$ be a unary CQ and $\mathcal{I}$ an interpretation. An element $d \in \Delta^{\mathcal{I}}$ is an *answer to q in $\mathcal{I}$*, written $\mathcal{I} \models q(d)$, if $q(x) \to (\mathcal{I}, d)$. Now let $\mathcal{O}$ be a *DL-Lite* ontology and $\mathcal{A}$ an ABox. An individual $a \in \mathsf{ind}(\mathcal{A})$ is an *answer to q on $\mathcal{A}$ w.r.t. $\mathcal{O}$*, written $\mathcal{A}, \mathcal{O} \models q(a)$, if a is an answer to q in every model of $\mathcal{O}$ and $\mathcal{A}$.

For q_1 and q_2 unary CQs and $\mathcal{O}$ a *DL-Lite* ontology, we say that q_1 is *contained in* q_2 w.r.t. $\mathcal{O}$, written $q_1 \subseteq_{\mathcal{O}} q_2$ if for all ABoxes $\mathcal{A}$ and $a \in \mathsf{ind}(\mathcal{A})$, $\mathcal{A}, \mathcal{O} \models q_1(a)$ implies $\mathcal{A}, \mathcal{O} \models q_2(a)$. We call q_1 and q_2 equivalent w.r.t. $\mathcal{O}$, written $q_1 \equiv_{\mathcal{O}} q_2$, if $q_1 \subseteq_{\mathcal{O}} q_2$ and $q_2 \subseteq_{\mathcal{O}} q_1$.

$\mathcal{O}$-saturatedness and $\mathcal{O}$-minimality. The following two technical notions are used throughout this paper. A CQ q is *$\mathcal{O}$-saturated*, with $\mathcal{O}$ a *DL-Lite* ontology, if $\mathcal{A}_q, \mathcal{O} \models A(y)$ implies $A(y) \in q$ for all $y \in \mathsf{var}(q)$ and $A \in \mathsf{N_C}$. It is *$\mathcal{O}$-minimal* if there is no $S \subsetneq \mathsf{var}(q)$ such that $q \equiv_{\mathcal{O}} q|_S$ with $q|_S$ the restriction of q to the atoms that only contain variables in S.

The following is a consequence of the fact that acyclic CQs over *DL-Lite* ontologies can be answered in polynomial time [8].

Lemma 1. *Given an ELIQ q and a DL-Lite ontology $\mathcal{O}$, we can find in polynomial time an $\mathcal{O}$-saturated and $\mathcal{O}$-minimal ELIQ q' with $q \equiv_{\mathcal{O}} q'$.*

Universal Model. Let $\mathcal{O}$ be a *DL-Lite* ontology and $\mathcal{A}$ an ABox that is satisfiable w.r.t. $\mathcal{O}$. A *trace* for $\mathcal{A}$ and $\mathcal{O}$ is a sequence $t = aR_1 \dots R_n$, $n \geq 0$, such that $a \in \mathsf{ind}(\mathcal{A})$, the basic concepts $\exists R_1, \dots, \exists R_n$ occur in $\mathcal{O}$, $\mathcal{A}, \mathcal{O} \models \exists R_1(a)$, and $\mathcal{O} \models \exists R_i^- \sqsubseteq \exists R_{i+1}$ for $1 \leq i < n$. Let $\mathbf{T}$ denote the set of all traces for $\mathcal{A}$ and $\mathcal{O}$. Then the *universal model* of $\mathcal{A}$ and $\mathcal{O}$ is

$$\begin{aligned}\mathcal{U}_{\mathcal{A},\mathcal{O}} = \mathcal{A} \cup \{A(a) \mid \mathcal{A}, \mathcal{O} \models A(a)\} \cup \\ \{A(tR) \mid tR \in \mathbf{T} \text{ and } \mathcal{O} \models \exists R^- \sqsubseteq A\} \cup \{R(t, tR) \mid tR \in \mathbf{T}\}.\end{aligned}$$

For brevity, we write $\mathcal{U}_{q,\mathcal{O}}$ instead of $\mathcal{U}_{\mathcal{A}_q,\mathcal{O}}$ for any conjunctive query q.

3 Computing Frontiers in Polynomial Time

We show that for every ELIQ q and *DL-Lite* ontology $\mathcal{O}$ such that q is satisfiable w.r.t. $\mathcal{O}$, there is a frontier of polynomial size that can be computed in polynomial time. This generalizes a result from [11] for the case without ontologies. We also observe that the same is not true when *DL-Lite* is extended with conjunction, and that ELIQs can be characterized up to equivalence by polynomially many data examples in the presence of *DL-Lite* ontologies.

Definition 1. *A* frontier *of an ELIQ q w.r.t. a DL-Lite ontology $\mathcal{O}$ is a finite set of ELIQs $\mathcal{F}$ such that*

1. *$q \subseteq_{\mathcal{O}} q_F$ for all $q_F \in \mathcal{F}$;*
2. *$q_F \not\subseteq_{\mathcal{O}} q$ for all $q_F \in \mathcal{F}$;*
3. *for all ELIQs q' with $q \subseteq_{\mathcal{O}} q' \not\subseteq_{\mathcal{O}} q$, there is a $q_F \in \mathcal{F}$ with $q_F \subseteq_{\mathcal{O}} q'$.*

It is not hard to see that frontiers that are minimal w.r.t. set inclusion are unique up to equivalence of the ELIQs in them, that is, if $\mathcal{F}_1$ and $\mathcal{F}_2$ are minimal frontiers of q w.r.t. $\mathcal{O}$, then for every $q_F \in \mathcal{F}_1$, there is a $q'_F \in \mathcal{F}_2$ such that $q_F \equiv_{\mathcal{O}} q'_F$, and vice versa. The following is the main result of this section.

Theorem 1. *Let $\mathcal{O}$ be a DL-Lite ontology and q an ELIQ that is satisfiable w.r.t. $\mathcal{O}$. Then a frontier of q w.r.t. $\mathcal{O}$ can be computed in polynomial time.*

For proving Theorem 1, we first observe that we can concentrate on ontologies that are in normal form.

Lemma 2. *For every DL-Lite ontology $\mathcal{O}$, we can construct in polynomial time a DL-Lite ontology $\mathcal{O}'$ in normal form such that for every ELIQ q, a frontier of q w.r.t. $\mathcal{O}$ can be constructed in polynomial time given a frontier of q w.r.t. $\mathcal{O}'$.*

The normalization of $\mathcal{O}$ introduces fresh concept names $X_{\exists R}$ that represent the basic concept $\exists R$. In the proof of Lemma 2, we construct the frontier of q w.r.t. $\mathcal{O}'$ by replacing atoms $X_{\exists R}(x)$ with atoms $R(x, y)$, y a fresh variable.

Now we start with the proof of Theorem 1. Let $\mathcal{O}$ and $q(x)$ be as in the formulation of the theorem, $\mathcal{O}$ in normal form. By Lemma 1 and the fact that equivalent queries have the same frontiers, we can assume that q is $\mathcal{O}$-saturated and $\mathcal{O}$-minimal. To construct a frontier of q w.r.t. $\mathcal{O}$, we consider all ways to weaken q in a minimal way where weakening means to construct from q an ELIQ q' such that $q \subseteq_{\mathcal{O}} q'$ and $q' \not\subseteq_{\mathcal{O}} q$.

We start with some notation. We view the answer variable x of q as the root of the undirected tree G_q, thus imposing a direction on this tree which allows us to use notions for directed trees, such as successor, predecessor, and leaf, for the variables in q. Note that the imposed direction is unrelated to the direction of (inverse) roles in atoms in q. For every $z \in \mathsf{var}(q)$, we use q_z to denote the ELIQ obtained from q by taking the subtree of G_q rooted at z and making z the answer variable. For each variable $y \in \mathsf{var}(q)$, we define a set Γ_y of atoms in q that mention y and represent options that we have for weakening q. Formally, Γ_y contains

- all role atoms $R(y, z) \in q_y$ and
- all concept atoms $A(y) \in q$ such that
  (i) there is no $B(y) \in q$ with $\mathcal{O} \models B \sqsubseteq A$ and $\mathcal{O} \not\models A \sqsubseteq B$ and
  (ii) there is no $R(y, z) \in q$ with $\mathcal{O} \models \exists R \sqsubseteq A$.

Informally, we can weaken q by choosing a $y \in \mathsf{var}(q)$ and then removing a concept atom $A(y) \in \Gamma_y$ or a role atom $R(y, z) \in \Gamma_y$ as well as the subtree of q rooted at variable z. However, such removals alone are not enough to obtain a *minimal* weakening of q and must be accompanied by certain additions, as detailed below. Note that Conditions (i) and (ii) are needed to ensure that removing $A(y)$ indeed weakens q, that is, the resulting query is not equivalent to q w.r.t. $\mathcal{O}$.

We define a set of ELIQs $\mathcal{F}_q(y)$ for every variable $y \in \mathsf{var}(q)$, by induction on the codepth of y in q. The set $\mathcal{F}_q(x)$ ultimately obtained is a frontier of q w.r.t. $\mathcal{O}$. For every $y \in \mathsf{var}(q)$, set

$$\mathcal{F}_q(y) = \{q^\alpha(y) \mid \alpha \in \Gamma_y\}$$

where $q^\alpha(y)$ is constructed by starting with $q_y(y)$ and then doing the following:

1. if $\alpha = A(y)$, remove all atoms $B(y)$ with $\mathcal{O} \models A \equiv B$ (including α);
2. if $\alpha = R(y, z)$, remove α and all atoms of q_z. For each $q^\beta(z) \in \mathcal{F}_q(z)$, add a disjoint copy $\tilde{q}^\beta$ of q^β and the role atom $R(y, \tilde{z})$ where $\tilde{z}$ is the copy of z in $\tilde{q}^\beta$;
3. for each $S(y, z_1) \in q_y$ with $S(y, z_1) \neq \alpha$, add a disjoint copy q' of q and the role atom $S(y', z_1)$ where y' is the copy of y in q';

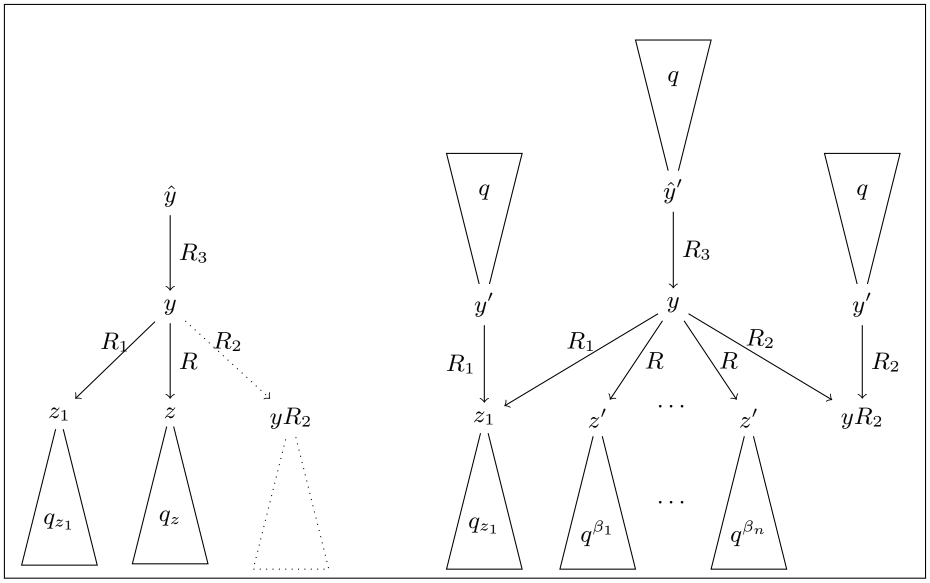

Fig. 1. Construction of $q^{R(y,z)}(y)$ from q

4. for each $S(y, yS) \in \mathcal{U}_{q_y,\mathcal{O}}$ with yS a trace such that $\mathcal{O} \not\models \exists S \sqsubseteq A$ if $\alpha = A(y)$, add a disjoint copy q' of q and the role atoms $S(y, z_1), S(y', z_1)$, where y' is the copy of y in q' and z_1 is a fresh variable name;
5. if there is a $S(\hat{y}, y) \in q$ with $\hat{y}$ the predecessor of y, then add a disjoint copy q' of q and the role atom $S(\hat{y}', y)$ where $\hat{y}'$ is the copy of $\hat{y}$ in q'.

Note that Step 2 is the inductive step, where every subtree rooted at a successor z of y is replaced with all ELIQs from $\mathcal{F}_q(z)$. The construction is illustrated in Figure 1 which on the left side shows a variable y in q with predecessor $\hat{y}$, two successors z_1 and z in q, and one additional successor yR_2 (a trace) in the universal model $\mathcal{U}_{q,\mathcal{O}}$. On the right side, it displays the ELIQ $q^{R(y,z)}(y) \in \mathcal{F}_q(y)$, assuming that $\mathcal{F}_q(z) = \{q^{\beta_1}, \ldots, q^{\beta_n}\}$. We remark that for $y \neq x$, the set $\mathcal{F}_q(y)$ is not necessarily a frontier of the ELIQ $q_y(y)$ because the part of q that is outside of subtree q_y is taken into account in the construction of $\mathcal{F}_q(y)$, in Steps 3-5. Our construction generalizes the construction of frontiers of ELIQs without ontologies given in [11].[3] It indeed yields a frontier.

Lemma 3. *$\mathcal{F}_q(x)$ is a frontier of $q(x)$ w.r.t. $\mathcal{O}$.*

We next observe that the obtained frontier $\mathcal{F}_q(x)$ is of polynomial size. It is then clear that it can be computed in polynomial time as described above since subsumption between basic concepts in *DL-Lite* can be decided in polynomial time [10].

[3] There is actually a small omission in [11] as the counterpart of our Step 5 is missing.

Lemma 4. $\sum_{q^\alpha(x)\in\mathcal{F}_q(x)} |\mathsf{var}(q^\alpha)| \leq |\mathsf{sig}(q)| \cdot |\mathsf{var}(q)|^3 \cdot (|\mathsf{var}(q)|+1) \cdot (||\mathcal{O}||+1).$

We next observe that adding conjunction to *DL-Lite* destroys polynomial frontiers and thus Theorem 1 does not extend to *DL-Lite*$_{\mathsf{horn}}$ ontologies [3]. In fact, this already holds for very simple queries and ontologies, implying that also for other DLs that support conjunction such as $\mathcal{EL}$, polynomial frontiers are elusive. A *conjunction of atomic queries (AQ^)* is a unary CQ of the form $q(x) \leftarrow A_1(x) \wedge \cdots \wedge A_n(x)$ and a *conjunctive ontology* is a set of CIs of the form $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq A$ where $A_1, \ldots, A_n$ and A are concept names.

Theorem 2. *There are families of AQ^s $q_1, q_2, \ldots$ and conjunctive ontologies $\mathcal{O}_1, \mathcal{O}_2, \ldots$ such that for all $n \geq 1$, any frontier of q_n w.r.t. $\mathcal{O}_n$ has size at least 2^n.*

The proof is a variation of a proof given in [15] showing that AQ^s are not polynomial time learnable under conjunctive ontologies. It is based on the following AQ^s and ontologies:

$$\begin{aligned} q_n(x) &\leftarrow A_1(x) \wedge A_1'(x) \wedge \cdots \wedge A_n(x) \wedge A_n'(x) \\ \mathcal{O}_n &= \{A_i \sqcap A_i' \sqsubseteq A_1 \sqcap A_1' \sqcap \cdots \sqcap A_n \sqcap A_n' \mid 1 \leq i \leq n\}. \end{aligned}$$

In fact, the minimal frontier contains all AQ^s that contain exactly one of the conjuncts A_i and A_i', for $1 \leq i \leq n$. Observe that in the proof of Theorem 1, there are only polynomially many choices for weakening an ELIQ, represented by the sets Γ_y, $y \in \mathsf{var}(q)$. In contrast, weakening the AQ^ q_n w.r.t. ontology $\mathcal{O}_n$ in a minimal way requires to choose for each $i \in \{1, \ldots, n\}$ whether A_i or A_i' should be removed, and there are exponentially many such choices.

To close this section, we briefly consider the unique characterization of ELIQs in terms of polynomially many data examples. A *data example* takes the form $(\mathcal{A}, a)$ where $\mathcal{A}$ is an ABox and $a \in \mathsf{ind}(\mathcal{A})$. Let E^+, E^- be finite sets of data examples. An ELIQ q *fits* (E^+, E^-) w.r.t. a *DL-Lite* ontology $\mathcal{O}$ if $(\mathcal{A}, a) \in E^+$ implies $\mathcal{A}, \mathcal{O} \models q(a)$ and $(\mathcal{A}, a) \in E^-$ implies $\mathcal{A}, \mathcal{O} \not\models q(a)$. We say that (E^+, E^-) *uniquely characterizes* q w.r.t. $\mathcal{O}$ if q fits (E^+, E^-) and every ELIQ q' that also fits (E^+, E^-) satisfies $q \equiv_{\mathcal{O}} q'$. The following is a consequence of Theorem 1.

Theorem 3. *For every DL-Lite ontology $\mathcal{O}$ and every ELIQ q that is satisfiable w.r.t. $\mathcal{O}$, we can compute in polynomial time data examples (E^+, E^-) that uniquely characterize q w.r.t. $\mathcal{O}$.*

Note that unique characterizability is closely related to the reverse engineering of CQs, also called *query-by-example* and studied in a DL context in [17,24].

4 Learning ELIQs

We use the results from the previous section to show that ELIQs are polynomial time learnable using only membership queries under *DL-Lite* ontologies if the learner is provided with an initial CQ q_H^0 such that q_H^0 is satisfiable w.r.t. the ontology $\mathcal{O}$ and $q_H^0 \subseteq_{\mathcal{O}} q_T$ where q_T is the target ELIQ. Such a q can be constructed in polynomial time if $\mathcal{O}$ is formulated in *DL-Lite*$^-$. Otherwise, it can be produced by a single initial equivalence query with an ELIQ that is not satisfiable w.r.t. $\mathcal{O}$, forcing the learner to provide a positive counterexample $(\mathcal{A}, a)$ from which we can extract the desired q_H^0. Before proving these positive results, however, we first observe that polynomial time learning using only membership queries (but no initial equivalence query) is not possible when $\mathcal{O}$ contains concept disjointness constraints.

A *disjointness ontology* is a *DL-Lite* ontology that only consists of concept disjointness constraints.

Theorem 4. *$AQ^\wedge$s are not polynomial query learnable under disjointness ontologies using only membership queries.*

The proof of Theorem 4 is a variation of that of Theorem 2. We next present the main results of this section.

Theorem 5.

1. *ELIQs are polynomial time learnable under DL-Lite ontologies using only membership queries and a single initial equivalence query.*
2. *ELIQs are polynomial time learnable under DL-Lite$^-$ ontologies using only membership queries.*

Throughout this section, we may assume the ontology to be in normal form.

Lemma 5. *If ELIQs are polynomial time learnable under DL-Lite ontologies in normal form using membership queries and a single initial equivalence query, then this is also true for unrestricted DL-Lite ontologies. The same holds for DL-Lite$^-$ ontologies without the initial equivalence query.*

The idea to prove Lemma 5 is to convert the given ontology into normal form and then run the learning algorithm for ontologies in normal form. Since that algorithm may pose membership queries and equivalence queries that involve fresh concept names introduced during normalization, we need to replace those concept names as described after Lemma 2 before forwarding the query to the oracle (which uses the original non-normalized ontology).

We prove Points 1 and 2 of Theorem 5 simultaneously. Let $\mathcal{O}$ be a *DL-Lite* ontology and Σ a finite signature that contains all symbols in $\mathcal{O}$, both known to the learner and the oracle. Further let $q_T(y)$ be the target ELIQ known to the oracle, formulated in signature Σ and satisfiable w.r.t. $\mathcal{O}$. The algorithm that enables the learner to learn q_T in polynomial time is displayed as Algorithm 1. It takes as input a CQ q_H^0 that is satisfiable w.r.t. $\mathcal{O}$ and satisfies $q_H^0 \subseteq_{\mathcal{O}} q_T$. Note that q_H^0 need not be an ELIQ. The algorithm then constructs and repeatedly updates a hypothesis ELIQ q_H while maintaining that $q_H \subseteq_{\mathcal{O}} q_T$. The initial call to subroutine $\mathsf{treeify}$ yields an ELIQ q_H with $q_H^0 \subseteq_{\mathcal{O}} q_H \subseteq_{\mathcal{O}} q_T$ to be used as the first hypothesis. The algorithm then iteratively generalizes q_H by constructing the frontier $\mathcal{F}_{q_H}$ of q_H w.r.t. $\mathcal{O}$ in polynomial time and choosing from it a new ELIQ q_H with $q_H \subseteq_{\mathcal{O}} q_T$. Additionally, the algorithm applies the $\mathsf{minimize}$ subroutine

Algorithm 1 Algorithm for learning ELIQs under *DL-Lite* ontologies

Input A *DL-Lite* ontology $\mathcal{O}$ and a CQ q_H^0 satisfiable w.r.t. $\mathcal{O}$ such that $q_H^0 \subseteq_{\mathcal{O}} q_T$
Output An ELIQ q_H such that $q_H \equiv_{\mathcal{O}} q_T$

$q_H := \mathsf{treeify}(q_H^0)$
while there is a $q_F \in \mathcal{F}_{q_H}$ with $q_F \subseteq_{\mathcal{O}} q_T$ **do**
  $q_H := \mathsf{minimize}(q_F)$
end while
return q_H

to ensure that the new q_H is $\mathcal{O}$-minimal and to avoid an excessive blowup while iterating in the while loop.

Before we detail the subroutines treeify and minimize, we explain how to obtain the argument q_H^0 to the algorithm. Suppose first that $\mathcal{O}$ contains neither concept disjointness constraints nor role disjointness constraints. Then

$$q_H^0(x) = \{A(x) \mid A \in \Sigma \cap \mathsf{N_C}\} \cup \{r(x,x) \mid r \in \Sigma \cap \mathsf{N_R}\}. \tag{1}$$

If $\mathcal{O}$ contains concept or role disjointness constraints, then we cannot use the above q_H^0 because it is not satisfiable w.r.t. $\mathcal{O}$. If, however, $\mathcal{O}$ contains only role disjointness constraints, then we can still find a suitable q_H^0. Let $\mathbf{R}$ be the set of all $r \in \Sigma \cap \mathsf{N_R}$ such that $\exists r$ is satisfiable w.r.t. $\mathcal{O}$, introduce four variables $x_r^0, x_r^1, x_{r^-}^0, x_{r^-}^1$ for all $r \in \mathbf{R}$, and fix a linear order $\preceq$ on $\mathbf{R}' = \mathbf{R} \cup \{r^- \mid r \in \mathbf{R}\}$. Fix any variable $x := x_R^i$. Then, q_H^0 is given by

$$\begin{aligned} q_H^0(x) = {} & \{A(x_R^i) \mid A \in \Sigma \cap \mathsf{N_C}, R \in \mathbf{R}', i \in \{0,1\}\} \cup \\ & \{R(x_S^i, x_R^i) \mid R, S \in \mathbf{R}', S \preceq R, i \in \{0,1\}\} \cup \\ & \{R(x_S^i, x_R^{1-i}) \mid R, S \in \mathbf{R}', S \not\preceq R, i \in \{0,1\}\}. \end{aligned}$$

Observe that every variable has an R-successor for every (satisfiable) role R. Therefore, there is a homomorphism from every satisfiable target ELIQ q_T to q_H^0, which shows that indeed $q_H^0 \subseteq_{\mathcal{O}} q_T$.

In the remaining case when $\mathcal{O}$ contains a concept disjointness constraint $A \sqcap B \sqsubseteq \bot$, we pose the ELIQ $A(x) \wedge B(x)$ as an equivalence query to the oracle. Since the target query is satisfiable w.r.t. $\mathcal{O}$, the oracle returns a positive counterexample $(\mathcal{A}, a)$. The desired query q_H^0 is $(\mathcal{A}, a)$ viewed as a CQ with answer variable a. In the algorithm, we may w.l.o.g. assume that q_H^0 is $\mathcal{O}$-saturated due to Lemma 1.

The minimize subroutine. The subroutine takes as input a unary CQ $q(x)$ that is $\mathcal{O}$-saturated, satisfiable w.r.t. $\mathcal{O}$, and satisfies $q \subseteq_{\mathcal{O}} q_T$. It computes an $\mathcal{O}$-minimal unary CQ q' with $q \subseteq_{\mathcal{O}} q' \subseteq_{\mathcal{O}} q_T$ using membership queries. This is done by exhaustively applying the following operation:

Remove successor. Choose a role atom $r(x_1, x_2) \in q$ and let q^- be the restriction of $q \setminus \{r(x_1, x_2)\}$ to the atoms that only contain variables which are reachable from x in $G_{q\setminus\{r(x_1,x_2)\}}$. Pose the membership query $\mathcal{A}_{q^-}, \mathcal{O} \models q_T(x)$. If the response is positive, continue with q^- in place of q.

This operation preserves $\mathcal{O}$-saturation and satisfiability w.r.t. $\mathcal{O}$. Since the operation is applied at most once to each atom in q, the running time and number of membership queries is polynomial in $|\mathsf{var}(q)| + |\Sigma|$. The following lemma summarizes important properties of $q' = \mathsf{minimize}(q)$ that we need to show correctness and polynomial running time of Algorithm 1.

Lemma 6. *Let q be a unary CQ that is $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$ such that $q \subseteq q_T$ for the target query $q_T(y)$, and let $q' = \mathsf{minimize}(q)$. Then*

1. *$q \subseteq_{\mathcal{O}} q'$ and $q' \subseteq_{\mathcal{O}} q_T$;*
2. *$\mathsf{var}(q') \subseteq \mathsf{img}(h)$ for every homomorphism h from q_T to $\mathcal{U}_{q',\mathcal{O}}$ with $h(y) = x$ (and consequently $|\mathsf{var}(q')| \leq |\mathsf{var}(q_T)|$);*
3. *q' is $\mathcal{O}$-minimal.*

When q is an ELIQ, $\mathsf{minimize}(q)$ is also an ELIQ. We define $\mathsf{minimize}$ on unrestricted CQs since it is applied to non-ELIQs as part of the $\mathsf{treeify}$ subroutine, described next.

The $\mathsf{treeify}$ subroutine. The subroutine takes as input a unary CQ $q(x)$ that is $\mathcal{O}$-saturated, satisfiable w.r.t. $\mathcal{O}$, and satisfies $q \subseteq_{\mathcal{O}} q_T$. It computes an ELIQ $q' = \mathsf{treeify}(q)$ with $q \subseteq_{\mathcal{O}} q' \subseteq_{\mathcal{O}} q_T$ by repeatedly increasing the length of cycles in q and posing membership queries; similar constructions are used in in [21,15]. Formally, $\mathsf{treeify}$ constructs a sequence of CQs $p_1, p_2, \ldots$ starting with $p_1 = \mathsf{minimize}(q)$ and then taking $p_{i+1} = \mathsf{minimize}(p'_i)$ where p'_i is obtained from p_i by doubling the length of every cycle. More precisely, p'_i is the result of the following operation.

Double cycle. Choose a cycle $R_1(x_1, x_2), \ldots, R_n(x_n, x_1)$ in p_i and let p'_i be the CQ obtained as follows: start with p_i, introduce copies $x'_1, \ldots, x'_n$ of $x_1, \ldots, x_n$, and

- remove all atoms $R(x_n, x_1)$;
- add $A(x'_j)$ if $A(x_j) \in p_i$ with $1 \leq j \leq n$;
- add $R(x'_j, z)$ if $R(x_j, z) \in p_i$ with $1 \leq j \leq n$ and $z \notin \{x_1, \ldots, x_n\}$;
- add $R(x'_j, x'_k)$ if $R(x_j, x_k) \in p_i$ with $1 \leq j, k \leq n$ and $\{j, k\} \neq \{1, n\}$;
- add $R(x_n, x'_1)$ and $R(x'_n, x_1)$ if $R(x_n, x_1) \in p_i$.

Once p_i contains no more cycles, $\mathsf{treeify}$ stops and returns p_i. The next lemma states the central properties of this construction.

Lemma 7. *For all $i \geq 1$,*

1. *p_i is $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$;*
2. *$p_i \subseteq_{\mathcal{O}} q_T$;*
3. *$p_i \subseteq_{\mathcal{O}} p_{i+1}$ and $|\mathsf{var}(p_{i+1})| > |\mathsf{var}(p_i)|$.*

Point 3 of Lemma 7 and the 'consequently' part of Point 2 of Lemma 6 imply that $\mathsf{treeify}$ terminates and thus eliminates all cycles in q while maintaining $q \subseteq_{\mathcal{O}} q_T$. The next lemma makes this precise.

Lemma 8. *Let q be a CQ that is $\mathcal{O}$-saturated, satisfiable w.r.t. $\mathcal{O}$, and satisfies $q \subseteq_{\mathcal{O}} q_T$. Then $q' = \mathsf{treeify}(q)$ is an ELIQ that can be computed in time polynomial in $|\mathsf{var}(q_T)| + ||q||$ using membership queries.*

Returning to Algorithm 1, let $q_1, q_2, \ldots$ be the sequence of ELIQs that are assigned to q_H during a run of the learning algorithm. Using the properties of frontiers, $\mathsf{minimize}$, and $\mathsf{treeify}$ we can now show that the hypotheses approximate the target query in an increasingly closer way.

Lemma 9. *For all $i \geq 1$,*

1. $q_i \subseteq_{\mathcal{O}} q_T$;
2. $q_i \subseteq_{\mathcal{O}} q_{i+1}$ *and* $q_{i+1} \not\subseteq_{\mathcal{O}} q_i$;
3. $\mathsf{var}(q_i) \subseteq \mathsf{img}(h)$ *for every homomorphism h from q_{i+1} to $\mathcal{U}_{q_i,\mathcal{O}}$ with $h(x) = x$.*

Point 3 of Lemma 9 implies that $|\mathsf{var}(q_{i+1})| \geq |\mathsf{var}(q_i)|$, and this can be used to show that the while loop in Algorithm 1 terminates after a polynomial number of iterations, arriving at a hypothesis $q_H \subseteq_{\mathcal{O}} q_T$ such that there is no $q_F \in \mathcal{F}_{q_H}$ with $q_F \subseteq_{\mathcal{O}} q_T$, that is, $q_H \equiv_{\mathcal{O}} q_T$.

Lemma 10. *$q_n \equiv q_T$ for some $n \leq p(|\mathsf{var}(q_T)| + |\Sigma|)$, p a polynomial.*

It follows from Lemma 10, the 'consequently' part of Point 2 of Lemma 6, and Lemma 8 that Algorithm 1 is a polynomial time learning algorithm, thus completing the proof of Theorem 5.

5 Outlook

As future work, we are going to consider extensions of the setup studied in this paper. We are optimistic that the approach presented here can be extended to *DL-Lite* ontologies with role inclusions, yielding polynomial size frontiers and polynomial query learnability also for that logic (but not polynomial time learnability since subsumption becomes NP-complete). Natural next steps could then be to replace *DL-Lite* with linear tuple-generating dependencies (TGDs) [9] or with $\mathit{DL\text{-}Lite}_{\mathsf{krom}}$ ontologies [3]. In contrast, it is clear from Theorem 2 and the results in [15] that our results do not extend to $\mathit{DL\text{-}Lite}_{\mathsf{horn}}$. It is not ruled out, however, that ELIQs can be learned in polynomial time w.r.t. $\mathit{DL\text{-}Lite}_{\mathsf{horn}}$ ontologies when both membership and equivalence queries can be used. Another natural question is whether CQs can be learned in polynomial time in the presence of *DL-Lite* ontologies. It is known that this is not possible with only membership queries even without ontologies [11], but it might be possible with both membership and equivalence queries.

Acknowledgement. Carsten Lutz was supported by DFG CRC 1320 EASE.

References

1. Angluin, D.: Learning regular sets from queries and counterexamples. Inf. Comput. **75**(2), 87–106 (1987)
2. Angluin, D.: Queries and concept learning. Mach. Learn. **2**(4), 319–342 (1987)
3. Artale, A., Calvanese, D., Kontchakov, R., Zakharyaschev, M.: The DL-Lite family and relations. J. of Artifical Intelligence Research **36**, 1–69 (2009)
4. Baader, F.: Least common subsumers and most specific concepts in a description logic with existential restrictions and terminological cycles. In: Proc. of IJCAI. pp. 319–324. Morgan Kaufmann (2003)
5. Baader, F., Horrocks, I., Lutz, C., Sattler, U.: An Introduction to Description Logics. Cambride University Press (2017)
6. Baader, F., Küsters, R., Molitor, R.: Computing least common subsumers in description logics with existential restrictions. In: Proc. of IJCAI. pp. 96–103. Morgan Kaufmann (1999)
7. Baader, F., Sertkaya, B., Turhan, A.: Computing the least common subsumer w.r.t. a background terminology. J. Appl. Log. **5**(3), 392–420 (2007)
8. Bienvenu, M., Ortiz, M., Simkus, M., Xiao, G.: Tractable queries for lightweight description logics. In: Proc. of IJCAI. pp. 768–774 (2013)
9. Calì, A., Gottlob, G., Lukasiewicz, T.: A general datalog-based framework for tractable query answering over ontologies. J. Web Semant. **14**, 57–83 (2012)
10. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tractable reasoning and efficient query answering in description logics: The DL-Lite family. J. of Automated Reasoning **39**(3), 385–429 (2007)
11. ten Cate, B., Dalmau, V.: Conjunctive queries: Unique characterizations and exact learnability. In: Proc. of ICDT. LIPIcs, vol. 186, pp. 9:1–9:24. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2021)
12. Cohen, W.W., Hirsh, H.: The learnability of description logics with equality constraints. Mach. Learn. **17**(2-3), 169–199 (1994)
13. Cohen, W.W., Hirsh, H.: Learning the classic description logic: Theoretical and experimental results. In: Proc. of KR. pp. 121–133. Morgan Kaufmann (1994)
14. Frazier, M., Pitt, L.: Classic learning. Mach. Learn. **25**(2-3), 151–193 (1996)
15. Funk, M., Jung, J.C., Lutz, C.: Actively learning concept and conjunctive queries under $\mathcal{EL}^r$-ontologies. In: Proc. of IJCAI (2021)
16. Funk, M., Jung, J.C., Lutz, C., Pulcini, H., Wolter, F.: Learning description logic concepts: When can positive and negative examples be separated? In: Proc. of IJCAI. pp. 1682–1688 (2019)
17. Gutiérrez-Basulto, V., Jung, J.C., Sabellek, L.: Reverse engineering queries in ontology-enriched systems: The case of expressive Horn description logic ontologies. In: Proc. of IJCAI-ECAI. ijcai.org (2018)
18. Jung, J.C., Lutz, C., Pulcini, H., Wolter, F.: Logical separability of incomplete data under ontologies. In: Proc. of KR. pp. 517–528 (2020)
19. Jung, J.C., Lutz, C., Wolter, F.: Least general generalizations in description logic: Verification and existence. In: Proc. of AAAI (2020)
20. Konev, B., Lutz, C., Ozaki, A., Wolter, F.: Exact learning of lightweight description logic ontologies. J. Mach. Learn. Res. **18**(201), 1–63 (2018)
21. Konev, B., Ozaki, A., Wolter, F.: A model for learning description logic ontologies based on exact learning. In: Proc. of AAAI. pp. 1008–1015. AAAI Press (2016)
22. Lehmann, J., Hitzler, P.: Concept learning in description logics using refinement operators. Mach. Learn. **78**, 203–250 (2010)

23. Lehmann, J., Völker, J.: Perspectives on Ontology Learning, Studies on the Semantic Web, vol. 18. IOS Press (2014)
24. Ortiz, M.: Ontology-mediated queries from examples: a glimpse at the DL-Lite case. In: Proc. of GCAI. pp. 1–14 (2019)
25. Ozaki, A.: Learning description logic ontologies: Five approaches. where do they stand? KI - Künstliche Intelligenz (2020)
26. Ozaki, A., Persia, C., Mazzullo, A.: Learning query inseparable $\mathcal{ELH}$ ontologies. In: Proc. of AAAI. pp. 2959–2966 (2020)
27. Sarker, M.K., Hitzler, P.: Efficient concept induction for description logics. In: Proc. of AAAI. pp. 3036–3043 (2019)
28. Zarrieß, B., Turhan, A.: Most specific generalizations w.r.t. general $\mathcal{EL}$-TBoxes. In: Proc. of IJCAI. pp. 1191–1197 (2013)

A Additional Preliminaries

The most important properties of universal models are as follows.

Lemma 11. *Let $\mathcal{O}$ be a DL-Lite ontology and $\mathcal{A}$ an ABox that is satisfiable w.r.t. $\mathcal{O}$. Then*

1. *$\mathcal{U}_{\mathcal{A},\mathcal{O}}$ is a model of $\mathcal{A}$ and $\mathcal{O}$;*
2. *$\mathcal{A},\mathcal{O} \models q(\bar{a})$ iff $q(\bar{x}) \to (\mathcal{U}_{\mathcal{A},\mathcal{O}},\bar{a})$ for all CQs $q(\bar{x})$ and all $\bar{a} \in \mathsf{ind}(\mathcal{A})^{|\bar{x}|}$.*

Lemma 12. *Let $\mathcal{O}$ be a DL-Lite ontology and $q_1(\bar{x})$, $q_2(\bar{y})$ be CQs that are satisfiable w.r.t. $\mathcal{O}$. Then $q_1 \subseteq_{\mathcal{O}} q_2$ iff $q_2(\bar{y}) \to (\mathcal{U}_{q_1,\mathcal{O}},\bar{x})$.*

Lemma 13. *Let $\mathcal{O}$ be a DL-Lite ontology, $\mathcal{A}$ an ABox, and $q(x)$ a unary CQ, such that $\mathcal{A}$ and q are both satisfiable w.r.t. $\mathcal{O}$. If h is a homomorphism from q to $\mathcal{U}_{\mathcal{A},\mathcal{O}}$ with $h(x) = a$ for some $a \in \mathsf{ind}(\mathcal{A})$, then there is a homomorphism h' from $\mathcal{U}_{q,\mathcal{O}}$ to $\mathcal{U}_{\mathcal{A},\mathcal{O}}$ with $h'(x) = a$ and $h'(x'R_1 \dots R_n) = h(x')R_1 \dots R_n$ for all traces $x'R_1 \dots R_n \in \mathsf{ind}(\mathcal{U}_{q,\mathcal{O}})$*

Definition 2. *An* $\mathcal{ELI}$-simulation *from interpretation $\mathcal{I}_1$ to interpretation $\mathcal{I}_2$ is a relation $S \subseteq \Delta^{\mathcal{I}_1} \times \Delta^{\mathcal{I}_2}$ such that for all $(d_1,d_2) \in S$, we have:*

1. *for all $A \in \mathsf{N_C}$: if $A(d_1) \in \mathcal{I}_1$, then $A(d_2) \in S$;*
2. *for all $r \in \mathsf{N_R}$ and $R \in \{r, r^-\}$: if there is some $d_1' \in \Delta^{\mathcal{I}_1}$ with $R(d_1,d_1') \in \mathcal{I}_1$, then there is $d_2' \in \Delta^{\mathcal{I}_2}$ such that $(d_1',d_2') \in S$ and $R(d_2,d_2') \in \mathcal{I}_2$.*

The following is a standard property of $\mathcal{ELI}$-simulations, we omit the proof.

Lemma 14. *Let $\mathcal{O}$ be a DL-Lite ontology, $\mathcal{A}_1$, $\mathcal{A}_2$ ABoxes and $q(x)$ an ELIQ such that $\mathcal{A}_1$, $\mathcal{A}_2$, and q are satisfiable w.r.t $\mathcal{O}$. If there is an $\mathcal{ELI}$-simulation S from $\mathcal{A}_1$ to $\mathcal{A}_2$ with $(a_1,a_2) \in S$, then $\mathcal{A}_1,\mathcal{O} \models q(a_1)$ implies $\mathcal{A}_2,\mathcal{O} \models q(a_2)$.*

Lemma 1. *Given an ELIQ q and a DL-Lite ontology $\mathcal{O}$, we can find in polynomial time an $\mathcal{O}$-saturated and $\mathcal{O}$-minimal ELIQ q' with $q \equiv_{\mathcal{O}} q'$.*

Proof. Let $\mathcal{O}$ be a *DL-Lite* ontology and $q(x)$ an ELIQ. We assume that $\mathcal{O}$ does not contain role disjointness constraints. Indeed, one can verify that, if $\mathcal{O}'$ is the result of dropping all role disjointness constraints from $\mathcal{O}$, we have that q is $\mathcal{O}$-saturated iff it is $\mathcal{O}'$-saturated and $\mathcal{O}$-minimal iff it is $\mathcal{O}'$-minimal.

For establishing $\mathcal{O}$-saturatedness, enumerate all atoms $A(y)$ with $y \in \mathsf{var}(q)$ and A a concept name that occurs in $\mathcal{O}$ and test whether $\mathcal{A}_q,\mathcal{O} \models A(y)$. If so, add $A(y)$ to the query. The test $\mathcal{A}_q,\mathcal{O} \models A(y)$ can be evaluated in polynomial time since evaluation of acyclic queries over *DL-Lite* ontologies without role disjointness constraints is in PTime [8].

For $\mathcal{O}$-minimality, enumerate all variables $y \in \mathsf{var}(q) \setminus \{x\}$, x the answer variable of q. For every such variable y, test whether $q|_S \subseteq_{\mathcal{O}} q$ where $q|_S$ is the restriction of q to the variables in $S = \mathsf{var}(q) \setminus \{y\}$. If so, replace q with $q|_S$. It is readily seen that the result is $\mathcal{O}$-minimal. Moreover, by Lemmas 12 and 11, the test $q|_S \subseteq_{\mathcal{O}} q$ is equivalent to the test $\mathcal{A}_{q|_S},\mathcal{O} \models q(x)$. The latter can be evaluated in polynomial time again since evaluation of acyclic queries over *DL-Lite* ontologies without role disjointness constraints is in PTime [8].

It remains to note that the two properties do not interact, that is, we can first establish $\mathcal{O}$-saturatedness and then $\mathcal{O}$-minimality. ❑

A *DL-Lite* ontology $\mathcal{O}_2$ is a *conservative extension* of a *DL-Lite* ontology $\mathcal{O}_1$ if $\mathsf{sig}(\mathcal{O}_1) \subseteq \mathsf{sig}(\mathcal{O}_2)$, every model of $\mathcal{O}_2$ is a model of $\mathcal{O}_1$, and for every model $\mathcal{I}_1$ of $\mathcal{O}_1$, there exists a model $\mathcal{I}_2$ of $\mathcal{O}_2$ such that $S^{\mathcal{I}_1} = S^{\mathcal{I}_2}$ for all symbols in $S \notin \mathsf{sig}(\mathcal{O}_2) \setminus \mathsf{sig}(\mathcal{O}_1)$.

Lemma 15. *Given a DL-Lite ontology $\mathcal{O}$, one can compute in polynomial time a DL-Lite ontology $\mathcal{O}'$ in normal form such that:*

1. *$\mathcal{O}'$ is a conservative extension of $\mathcal{O}$,*
2. $\mathsf{sig}(\mathcal{O}') = \mathsf{sig}(\mathcal{O}) \cup \{X_{\exists R} \mid \exists R$ *occurs in* $\mathcal{O}\}$,
3. *$\mathcal{O}' \models X_{\exists R} \equiv \exists R$ for each $\exists R$ that occurs in $\mathcal{O}$.*

Proof. Introduce a fresh concept name X_B for every basic concept B of the form $\exists R$ in $\mathcal{O}$. Then $\mathcal{O}'$ is obtained from $\mathcal{O}$ by first replacing every basic concept of the form $\exists R$ by $X_{\exists R}$, and then adding the concept inclusions $\exists R \sqsubseteq X_{\exists R}$ and $X_{\exists R} \sqsubseteq \exists R$ to $\mathcal{O}'$. $\mathcal{O}'$ is in normal form and can be computed in polynomial time. Moreover, it can be verified that Points 1–3 hold. ❑

B Proofs for Section 3

Lemma 2. *For every DL-Lite ontology $\mathcal{O}$, we can construct in polynomial time a DL-Lite ontology $\mathcal{O}'$ in normal form such that for every ELIQ q, a frontier of q w.r.t. $\mathcal{O}$ can be constructed in polynomial time given a frontier of q w.r.t. $\mathcal{O}'$.*

Proof. Let $\mathcal{O}$ be a *DL-Lite* ontology. Let $\mathcal{O}'$ be the ontology in normal form that is computed in Lemma 15. Moreover, let $q(x)$ be an ELIQ and $\mathcal{F}$ be a frontier of q w.r.t. $\mathcal{O}$'. Obtain $\mathcal{F}'$ from $\mathcal{F}$ by replacing every occurrence of an atom $X_{\exists R}(y)$ in some query in $\mathcal{F}$ with an atom $R(y, z)$, using a fresh z for every replacement. We verify that $\mathcal{F}'$ is a frontier of q w.r.t. $\mathcal{O}$:

1. Suppose there is $p(x) \in \mathcal{F}'$ with $q \not\subseteq_{\mathcal{O}} p$. Then, there is an ABox $\mathcal{A}$ and an individual $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A}, \mathcal{O} \models q(a)$, but $\mathcal{A}, \mathcal{O} \not\models p(a)$. By construction, of $\mathcal{O}'$, we have $\mathcal{A}, \mathcal{O}' \models q(a)$. Since $\mathcal{A}, \mathcal{O} \not\models p(a)$, there is a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{O}$ such that $\mathcal{I} \not\models p(a)$. Now, $\mathcal{I}$ can be extended to a model of $\mathcal{A}$ and $\mathcal{O}'$ by taking $X_{\exists R}^{\mathcal{I}} = (\exists R)^{\mathcal{I}}$. Now, let $p_0(x) \in \mathcal{F}$ be the query from which $p(x)$ was obtained. Clearly, the extended $\mathcal{I}$ satisfies $\mathcal{I} \not\models p_0(a)$, so $q \not\subseteq_{\mathcal{O}'} p$, a contradiction.
2. Let $p(x)$ be an element of $\mathcal{F}'$ and $p_0(x) \in \mathcal{F}$ be the query from which $p(x)$ was obtained. By Point 2 of the definition of a frontier, $p_0 \not\subseteq_{\mathcal{O}'} q$, that is, there is an ABox $\mathcal{A}$ and an individual $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A}, \mathcal{O}' \models p_0(a)$, but $\mathcal{A}, \mathcal{O}' \not\models q(a)$. From the former we obtain $\mathcal{A}, \mathcal{O} \models p(a)$, since each model of $\mathcal{A}$ and $\mathcal{O}$ can be extended to a model of $\mathcal{O}'$. From the latter, we obtain that there is a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{O}'$ such that $\mathcal{I} \not\models q(a)$. But $\mathcal{I}$ is also a model of $\mathcal{O}$, thus $\mathcal{A}, \mathcal{O} \not\models q(a)$. Overall, we have $p \not\subseteq_{\mathcal{O}} q$.

3. Let $q'(x)$ be an ELIQ with $q \subseteq_{\mathcal{O}} q' \not\subseteq_{\mathcal{O}} q$. Since $\mathcal{O}$' is a conservative extension of $\mathcal{O}$ and q' does not use the symbols introduced in the construction of $\mathcal{O}'$, we also have $q \subseteq_{\mathcal{O}'} q' \not\subseteq_{\mathcal{O}'} q$. Let $p_0(x) \in \mathcal{F}$ witness Point 3 of the definition of a frontier for q', and let $p(x) \in \mathcal{F}'$ be obtained from $p_0(x)$. If $p_0 \not\subseteq_{\mathcal{O}} q'$, we can use the counter model as in the previous points to show that $p \not\subseteq_{\mathcal{O}'} q'$, a contradiction.

❏

Lemma 16. *Let $\mathcal{O}$ be a DL-Lite ontology in normal form and let $q(x)$ be an ELIQ that is $\mathcal{O}$-minimal, $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$. Then $\mathsf{var}(q) \subseteq \mathsf{img}(h)$ for every homomorphism h from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h(x) = x$.*

Proof. Assume for contradiction that there is a variable $x' \in \mathsf{var}(q)$ with $x' \notin \mathsf{img}(h)$. Let q' be the restriction of q to $\mathsf{var}(q) \setminus \mathsf{var}(q_{x'})$. We show that h is also a homomorphism from q to $\mathcal{U}_{q',\mathcal{O}}$, and thus that $q \equiv_{\mathcal{O}} q'$. This contradicts the minimality of q.

First, observe that by the normal form of $\mathcal{O}$ and the $\mathcal{O}$-saturation of q, the trace-subtrees below a variable $x_1 \in \mathsf{ind}(q')$ in $\mathcal{U}_{q',\mathcal{O}}$ and $\mathcal{U}_{q,\mathcal{O}}$ are identical.

Let $A(x_1) \in q$. Note that $h(x_1) \notin \mathsf{var}(q_{x'})$ since $x' \notin \mathsf{img}(h)$, q is connected, and $h(x) = x$. It thus follows from $\mathcal{O}$-saturation that $A(h(x_1)) \in \mathcal{U}_{q',\mathcal{O}}$. If $h(x_1)$ is a trace, then by the connectedness of q, it is a trace below a variable of q' and thus $A(h(x_1)) \in \mathcal{U}_{q',\mathcal{O}}$.

Let $r(x_1, x_2) \in q$. By $h(x_1), h(x_2) \notin \mathsf{var}(q_{x'})$ and the connectedness of q, both $h(x_1)$ and $h(x_2)$ must be variables of q' or traces starting with variables of q'. If both $h(x_1)$ and $h(x_2)$ are variables of q', then $r(h(x_1), h(x_2)) \in \mathcal{U}_{q',\mathcal{O}}$. If one or both of $h(x_1)$ and $h(x_2)$ is a trace, then since the trace-subtrees are identical, also $r(h(x_1), h(x_2)) \in \mathcal{U}_{q',\mathcal{O}}$. ❏

Note that this implies that every homomorphism h from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h(x) = x$ needs to be injective.

Lemma 3. *$\mathcal{F}_q(x)$ is a frontier of $q(x)$ w.r.t. $\mathcal{O}$.*

Proof. We show that $\mathcal{F}_q(x)$ fulfills the three conditions of frontiers. For Condition 1, let $q^\alpha(x) \in \mathcal{F}_q(x)$. Every variable in q^α is a copy of an element of $\mathcal{U}_{q,\mathcal{O}}$. The mapping $h\colon \mathsf{var}(q^\alpha) \to \mathsf{ind}(\mathcal{U}_{q,\mathcal{O}})$ that maps every copy to its original, from now on referred to as the natural projection, is a homomorphism from q^α to $\mathcal{U}_{q,\mathcal{O}}$ with $h(x) = x$. Thus $q \subseteq_{\mathcal{O}} q^\alpha$, as required.

For Condition 2, we show the following two claims:

Claim 1. For all $y \in \mathsf{var}(q)$ and $A(y) \in \Gamma_y$, $q^{A(y)} \not\subseteq_{\mathcal{O}} q_y$.

Proof of Claim 1. By Condition (i) of Γ_y and Step 1 of the construction of $q^{A(y)}$, there is no $B(y) \in q^{A(y)}$ with $\mathcal{O} \models B \sqsubseteq A$. Moreover, by Condition (ii) of Γ_y and the condition in Step 4 of the construction, there is also no $R(y, z) \in q^{A(y)}$ with $\mathcal{O} \models \exists R \sqsubseteq A$. Thus, $A(y) \in q_y$ but $A(y) \notin \mathcal{U}_{q^{A(y)},\mathcal{O}}$ and therefore $q_y \not\to (\mathcal{U}_{q^{A(y)},\mathcal{O}}, y)$. Hence, $q^{A(y)} \not\subseteq_{\mathcal{O}} q_y$ by Lemma 12.

Claim 2. For all $y \in \mathsf{var}(q)$ and $R(y,z) \in \Gamma_y$, $q^{R(y,z)} \not\subseteq_{\mathcal{O}} q_y$.

Proof of Claim 2. We show this claim by induction on the codepth of y. In the induction start, y has codepth 0. Then Claim 2 is trivially satisfied because there are no atoms $R(y,z) \in \Gamma_y$.

Thus, let y have codepth > 0 and assume that the claim holds for all smaller codepths. Assume for contradiction that there is a homomorphism h from q_y to $\mathcal{U}_{q^{R(y,z)},\mathcal{O}}$ with $h(y) = y$. Note that z has smaller codepth than y. By the induction hypothesis and Claim 1, $q_z(z) \not\to (\mathcal{U}_{q^\beta,\mathcal{O}}, z)$ for all $q^\beta \in \mathcal{F}_q(z)$. Since h must map z to a successor of y, we may distinguish the following cases.

If h maps z to some $\tilde{z}$ that was added in Step 2 of the construction and h maps q_z entirely into the subtree below $\tilde{z}$ in $\mathcal{U}_{q^{R(y,z)},\mathcal{O}}$, then let h' be the restriction of h to variables in q_z. Since the subtree below $\tilde{z}$ in $\mathcal{U}_{q^{R(y,z)},\mathcal{O}}$ is identical to $\mathcal{U}_{q^\beta,\mathcal{O}}$ for some $q^\beta \in \mathcal{F}_q(z)$, h' is also a homomorphism from q_z to $\mathcal{U}_{q^\beta,\mathcal{O}}$ with $h'(z) = z$. In the case of $\beta = A(z)$, this contradicts Claim 1, and in the case of $\beta = S(z,z')$ this contradicts the induction hypothesis.

The remaining cases imply that q is not $\mathcal{O}$-minimal, thus also leading to a contradiction. To show this, it is convenient to construct some homomorphisms beforehand. First, construct a homomorphism h' from q_y to $\mathcal{U}_{q^{R(y,z)},\mathcal{O}}$ with $h'(y) = y$ by setting $h(y') = y'$ for all $y' \notin \mathsf{var}(q_z)$ and $h'(y') = h(y')$ for all $y' \in \mathsf{var}(q_z)$. Then, compose h' with the natural projection from $\mathcal{U}_{q^{R(y,z)},\mathcal{O}}$ to $\mathcal{U}_{q,\mathcal{O}}$ to construct a homomorphism g from q_y to $\mathcal{U}_{q,\mathcal{O}}$ with $g(y) = y$. Finally, construct a homomorphism g' from q to $\mathcal{U}_{q,\mathcal{O}}$ with $g'(x) = x$ by extending g to be the identity on all variables not in $\mathsf{var}(q_y)$.

Continuing with the cases, next assume that h maps z to one of its copies $\tilde{z}$, but does not map q_z entirely into the subtree below $\tilde{z}$. Then there must be a $z_1 \in \mathsf{var}(q_z)$ such that $h(z_1) = y$. Thus h is not injective since $h(z_1) = y$ and $h(y) = y$. Then g' is also not injective, contradicting that q is $\mathcal{O}$-minimal by Lemma 16.

If h maps z to a variable z_1 added by Step 4 of the construction, then the natural projection maps z_1 to a trace yS. The homomorphism g' then also maps z to a trace in $\mathcal{U}_{q,\mathcal{O}}$, contradicting that q is $\mathcal{O}$-minimal by Lemma 16.

If h maps z to a trace yS, then g' maps z to a trace in $\mathcal{U}_{q,\mathcal{O}}$, again contradicting that q is $\mathcal{O}$-minimal.

If h maps z to a successor $z_2 \neq z$ of y, then h' is not injective since $h'(z) = z_2$ and $h'(z_2) = z_2$. Thus g' is not injective, contradicting that q is $\mathcal{O}$-minimal by Lemma 16.

If h maps z to the copy $\hat{y}'$ of the predecessor of y added by Step 5, then $g(z) = \hat{y}$ and thus $g'(z) = \hat{y}$ as well as $g'(\hat{y}) = \hat{y}$, again contradicting that q is $\mathcal{O}$-minimal by Lemma 16.

This completes the proof of Claim 2. Claim 1 and Claim 2 together imply Condition 2 of frontiers. For Condition 3 we show the following two claims:

Claim 3. Let $q'(x_1)$ be an ELIQ and $y \in \mathsf{var}(q)$ such that $q'(x_1) \to (\mathcal{U}_{q,\mathcal{O}}, y)$. If there is an $A(y) \in q_y$ such that $A(x_1) \notin \mathcal{U}_{q',\mathcal{O}}$, then there is a $q^\alpha(y) \in \mathcal{F}_q(y)$ such that $q'(x_1) \to (\mathcal{U}_{q^\alpha,\mathcal{O}}, y)$.

Proof of Claim 3. We assume that there is no $B(y) \in q_y$ with $\mathcal{O} \models B \sqsubseteq A$ and $\mathcal{O} \not\models A \sqsubseteq B$ (otherwise apply Claim 3 to $B(y)$) and that there is no $R(y,z) \in q_y$ with $\mathcal{O} \models \exists R \sqsubseteq A$ (otherwise Claim 4 applies to the atom $R(y,z)$). Hence $A(y)$ fulfills Conditions (i) and (ii) of Γ_y and there is a $q^{A(y)} \in \mathcal{F}_q(y)$. We construct a homomorphism h' from q' to $\mathcal{U}_{q^{A(y)},\mathcal{O}}$ with $h'(x_1) = y$ by by starting with $h'(x_1) = y$ and mapping the subtree below each successor x_2 of x_1 as follows. Let h be a homomorphism from q' to $\mathcal{U}_{q,\mathcal{O}}$ with $h(x_1) = y$.

If h maps x_2 to the predecessor $\hat{y}$ of y, set $h'(x_2) = \hat{y}'$ and continue mapping the subquery q'_{x_2} according to h into the copy of q added in Step 5 of the construction.

If h maps x_2 to a successor z_1 of y, set $h'(x_2) = z_1$ and continue mapping the subquery q'_{x_2} into the tree below z_1 according to h. If there is a $x_3 \in \mathsf{var}(q'_{x_2})$ with $h(x_3) = y$, instead set $h'(x_3) = y'$ and continue mapping q'_{x_3} into the copy of q added in Step 3 of the construction according to h.

If h maps x_2 to a trace yR, first note that by the choice of $A(y)$, $\mathcal{O} \not\models \exists R \sqsubseteq A$. Thus Step 4 of the construction of $q^{A(y)}$ added a copy z_1 of yR and the role atom $R(y, z_1)$. Set $h'(x_2) = z_1$ and continue mapping the subquery q'_{x_2} into the traces starting with z_1 according to h. If there is a $x_3 \in \mathsf{var}(q'_{x_2})$ with $h(x_3) = y$ set $h'(x_3) = y'$ and continue to map the subquery q'_{x_3} into the copy of q added in Step 4 according to h.

By construction, h' is a homomorphism. This completes the proof of Claim 3.

Claim 4. Let $q'(x_1)$ be an ELIQ and $y \in \mathsf{var}(q)$ such that $q'(x_1) \to (\mathcal{U}_{q,\mathcal{O}}, y)$. If there is a $R(y,z) \in q_y$ such that $q_z(z) \not\to (\mathcal{U}_{q',\mathcal{O}}, x_2)$ for all $R(x_1, x_2) \in \mathcal{U}_{q',\mathcal{O}}$, then there is a $q^\alpha(y) \in \mathcal{F}_q(y)$ such that $q'(x_1) \to (\mathcal{U}_{q^\alpha,\mathcal{O}}, y)$.

Proof of Claim 4. We show the claim by induction on the depth of q'. Start with a q' of depth 0, then there is no $R(x_1, x_2) \in q'$, but by assumption $R(y,z) \in q_y$. Hence there is a $q^{R(y,z)} \in \mathcal{F}_q(y)$. We construct a homomorphism h' from q' to $\mathcal{U}_{q^{R(y,z)},\mathcal{O}}$ by setting $h'(z) = y$. Since q is $\mathcal{O}$-saturated and the construction of $q^{R(y,z)}$ does not remove any concept atoms, $A(y) \in \mathcal{U}_{q,\mathcal{O}}$ iff $A(y) \in \mathcal{U}_{q^{R(y,z)},\mathcal{O}}$. Thus h' is a homomorphism.

Now let q' be an ELIQ of depth > 0 and assume that the claim holds for all ELIQs of smaller depth. Again, we construct a homomorphism h' from q' to $\mathcal{U}_{q^{R(y,z)},\mathcal{O}}$ with $h'(x_1) = y$. Start by setting $h'(x_1) = y$ and continue mapping the successors of x_1 as follows. Let h be a homomorphism from q' to $\mathcal{U}_{q,\mathcal{O}}$ with $h(x_1) = y$.

If h maps a successor x_2 to z, observe that q'_{x_2} is an ELIQ of smaller depth than q_z, that $q'_{x_2}(x_2) \to (\mathcal{U}_{q,\mathcal{O}}, z)$ and that $q_z(z) \not\to (\mathcal{U}_{q',\mathcal{O}}, x_2)$. Therefore either the induction hypothesis or Claim 3 can be applied to show that there is a $q^\beta \in \mathcal{F}_q(z)$ such that there is a homomorphism g from q'_{x_2} to $\mathcal{U}_{q^\beta,\mathcal{O}}$ with $g(x_2) = z$. Extend h' to the subquery q'_{x_2} by mapping it into the copy of q^β according to g.

For the remaining cases of h mapping a successor x_2 to the predecessor $\hat{y}$, to a trace starting with y, or to a successor of y that is not z, h' is expanded to the subquery q'_{x_2} as in the proof of Claim 3.

This completes the proof of Claim 4. Claim 3 and Claim 4 together imply that for all ELIQs $q'(x_1)$ and $y \in \mathsf{ind}(q)$ with $q'(x_1) \to (\mathcal{U}_{q,\mathcal{O}}, y)$ and $q_y(y) \not\to$

$(\mathcal{U}_{q',\mathcal{O}}, x_1)$, there is a $q^\alpha \in \mathcal{F}_q(y)$ such that $q'(x_1) \to (\mathcal{U}_{q^\alpha,\mathcal{O}}, y)$. Condition 3 of frontiers follows for $y = x$ by Lemma 12. ❑

Lemma 4. $\displaystyle\sum_{q^\alpha(x)\in\mathcal{F}_q(x)} |\mathsf{var}(q^\alpha)| \leq |\mathsf{sig}(q)| \cdot |\mathsf{var}(q)|^3 \cdot (|\mathsf{var}(q)| + 1) \cdot (||\mathcal{O}|| + 1).$

Proof. Let $q^\alpha \in \mathcal{F}_q(x)$. Recall that in Steps 3, 4, and 5, disjoint copies of q are added to q^α. Let $V(q^\alpha)$ denote the set of variables of q^α that are *not* part of such a copy of q, and let $N(q^\alpha)$ denote their number. Since at most one copy of q is added per variable in $V(q^\alpha)$, we have $|\mathsf{var}(q^\alpha)| \leq N(q^\alpha) + N(q^\alpha) \cdot |\mathsf{var}(q)| = N(q^\alpha) \cdot (\mathsf{var}(|q|) + 1)$. Hence, it suffices to show that

$$\sum_{q^\alpha\in\mathcal{F}_q(x)} N(q^\alpha) \leq |\mathsf{sig}(q)| \cdot |\mathsf{var}(q)|^3 \cdot (||\mathcal{O}|| + 1).$$

This bound follows from the following claim for $y = x$:

Claim. For all $y \in \mathsf{var}(q)$, we have:

$$\sum_{q^\alpha\in\mathcal{F}_q(y)} N(q^\alpha) \leq |\mathsf{sig}(q)| \cdot |\mathsf{var}(q_y)|^3 \cdot (||\mathcal{O}|| + 1).$$

Proof of the Claim. We show the claim by induction on the codepth of y in q. For the base case, consider a variable y of codepth 0. In that case $\mathcal{F}_q(x)$ contains only queries q^α for $\alpha = A(y)$ and $A(y) \in q$. Note that in the construction of q^α for $\alpha = A(y)$, only Step 4 introduces variables to $V(q^\alpha)$. More precisely, $V(q^\alpha)$ consists of $\mathsf{var}(q_y)$ and at most one variable for every atom $S(y, yS) \in \mathcal{U}_{q_y,\mathcal{O}}$. As the number of these atoms is bounded by $||\mathcal{O}||$, we have

$$N(q^{A(y)}) \leq |\mathsf{var}(q_y)| + ||\mathcal{O}||.$$

Since the number of atoms $A(y) \in q$ is bounded by $|\mathsf{sig}(q)|$, we obtain

$$\sum_{q^\alpha\in\mathcal{F}_q(y)} N(q^\alpha) \leq |\mathsf{sig}(q)| \cdot (|\mathsf{var}(q_y)| + ||\mathcal{O}||) \leq |\mathsf{sig}(q)| \cdot |\mathsf{var} q_y)|^3 \cdot (||\mathcal{O}|| + 1), \quad (2)$$

as required.

For the inductive step, consider a variable y of codepth greater than 0. By definition of $\mathcal{F}_q$, we have

$$\sum_{q^\alpha\in\mathcal{F}_q(y)} N(q^\alpha) = \sum_{A(y)\in\Gamma_y} N(q^{A(y)}) + \sum_{R(y,z)\in\Gamma_y} N(q^{R(y,z)}). \quad (3)$$

We bound from above the two sums on the right-hand side of (3). Using the exact same analysis as in the base case, one can show that the first sum is at most $|\mathsf{sig}(q)| \cdot (|\mathsf{var}(q_y)| + ||\mathcal{O}||)$, see (2). For the second sum, we analyze the construction of $q^{R(y,z)}$. Note first that z has smaller codepth than y, by definition of Γ_y. Moreover, Step 2 adds a copy of each $q^\beta \in \mathcal{F}_q(z)$ and Step 4 adds a variable to $V(q^{R(y,z)})$ for every atom $S(y, yS) \in \mathcal{U}_{q_y,\mathcal{O}}$. Since the number of the latter is bounded by $||\mathcal{O}||$ and using induction, we obtain

$$\begin{aligned}
N(q^{R(y,z)}) &\leq |\mathsf{var}(q_y)| + ||\mathcal{O}|| + \sum_{q^\beta \in \mathcal{F}_q(z)} N(q^\beta) \\
&\leq |\mathsf{var}(q_y)| + ||\mathcal{O}|| + |\mathsf{sig}(q)| \cdot |\mathsf{var}(q_z)|^3 \cdot (||\mathcal{O}|| + 1).
\end{aligned}$$

Now, the second sum in (3) can be bounded as follows

$$\begin{aligned}
\sum_{R(y,z)\in\Gamma_y} N(q^{R(y,z)}) &\leq \sum_{R(y,z)\in\Gamma_y} |\mathsf{var}(q_y)| + ||\mathcal{O}|| + |\mathsf{sig}(q)| \cdot |\mathsf{var}(q_z)|^3 \cdot (||\mathcal{O}|| + 1) \\
&\leq (|\mathsf{var}(q_y)| - 1) \cdot (|\mathsf{var}(q_y)| + ||\mathcal{O}||) \\
&\qquad + |\mathsf{sig}(q)| \cdot (||\mathcal{O}|| + 1) \cdot (\sum_{R(y,z)\in\Gamma_y} |\mathsf{var}(q_z)|)^3 \\
&\leq (|\mathsf{var}(q_y)| - 1) \cdot |\mathsf{var}(q_y)| \cdot (||\mathcal{O}|| + 1) \\
&\qquad + |\mathsf{sig}(q)| \cdot (||\mathcal{O}|| + 1) \cdot (|\mathsf{var}(q_y)| - 1)^3 \\
&\leq |\mathsf{sig}(q)| \cdot (||\mathcal{O}|| + 1) \cdot (|\mathsf{var}(q_y)|^2 - |\mathsf{var}(q_y)| + (|\mathsf{var}(q_y)| - 1)^3)
\end{aligned}$$

using $|\{R(y,z) \in \Gamma_y\}| \leq |\mathsf{var}(q_y)| - 1$ and $\sum_{j=1}^m f(j)^3 \leq (\sum_{j=1}^m f(j))^3$ for the second inequality and $\sum_{R(y,z)\in\Gamma_y} |\mathsf{var}(q_z)| = |\mathsf{var}(q_y)| - 1$ for the third.

To finish the proof of the claim, we put the bounds on the sums in (3) together and obtain that $\sum_{q^\alpha \in \mathcal{F}_q(y)} N(q^\alpha)$ is bounded from above by

$$\begin{aligned}
&|\mathsf{sig}(q)| \cdot (|\mathsf{var}(q_y)| + ||\mathcal{O}||) + \\
&\qquad |\mathsf{sig}(q)| \cdot (||\mathcal{O}|| + 1) \cdot (|\mathsf{var}(q_y)|^2 - |\mathsf{var}(q_y)| + (|\mathsf{var}(q_y)| - 1)^3) \\
&\leq |\mathsf{sig}(q)| \cdot (||\mathcal{O}|| + 1) \cdot (|\mathsf{var}(q_y)|^2 + (|\mathsf{var}(q_y)| - 1)^3) \\
&\leq |\mathsf{sig}(q)| \cdot (||\mathcal{O}|| + 1) \cdot |\mathsf{var}(q_y)|^3,
\end{aligned}$$

as required. ❏

Theorem 2. *There are families of AQ$^\wedge$s $q_1, q_2, \ldots$ and conjunctive ontologies $\mathcal{O}_1, \mathcal{O}_2, \ldots$ such that for all $n \geq 1$, any frontier of q_n w.r.t. $\mathcal{O}_n$ has size at least 2^n.*

Proof. For $n \geq 1$, let

$$\begin{aligned}
q_n(x) &\leftarrow A_1(x) \wedge A'_1(x) \wedge \cdots \wedge A_n(x) \wedge A'_n(x) \\
\mathcal{O}_n &= \{A_i \sqcap A'_i \sqsubseteq A_1 \sqcap A'_1 \sqcap \cdots \sqcap A_n \sqcap A'_n \mid 1 \leq i \leq n\}.
\end{aligned}$$

Suppose $\mathcal{F}$ is a frontier of q_n w.r.t. $\mathcal{O}_n$. Let p be any query that contains for each i with $1 \leq i \leq n$ either $A_i(x)$ or $A'_i(x)$. It suffices to show that $p \in \mathcal{F}$.

Clearly, $q_n \subseteq_{\mathcal{O}_n} p \not\subseteq_{\mathcal{O}_n} q_n$ and thus Point 3 of the definition of frontiers implies that there is a $p' \in \mathcal{F}$ with $p' \subseteq_{\mathcal{O}} p$. We distinguish cases:

– p' contains the atoms $A_i(x), A'_i(x)$ for some i. But then $p' \equiv_{\mathcal{O}_n} q_n$ and p' cannot be in $\mathcal{F}$ by Point 2 of the definition of frontiers, a contradiction.
– p' does not contain both atoms $A_i(x), A'_i(x)$ for any i. But then the ontology does not have an effect on the containment $p' \subseteq_{\mathcal{O}_n} p$ and hence every $A_i(x), A'_i(x)$ that occurs in p must occur in p'. As p' does not contain the atoms $A_i(x), A'_i(x)$ for any i, we actually have $p' = p$, which was to be shown.

❑

Theorem 3. *For every DL-Lite ontology $\mathcal{O}$ and every ELIQ q that is satisfiable w.r.t. $\mathcal{O}$, we can compute in polynomial time data examples (E^+, E^-) that uniquely characterize q w.r.t. $\mathcal{O}$.*

Proof. Let $q(x)$ and $\mathcal{O}$ be as in the Theorem. By Theorem 1, we can compute in polynomial time a frontier $\mathcal{F}_q(x)$ for q w.r.t. $\mathcal{O}$. Let $E^+ = \{(\mathcal{A}_q, x)\}$ and $E^- = \{(\mathcal{A}_p, x) \mid p \in \mathcal{F}_q(x)\}$. Let q' be an ELIQ that fits (E^+, E^-). Note first that we have $q \subseteq_{\mathcal{O}} q'$, since $(\mathcal{A}_q, x)$ is a positive example. Moreover, since all data examples in E^- are negative examples for q', we know that $p \not\subseteq_{\mathcal{O}} q'$ for any $p \in \mathcal{F}_q(x)$. By Point 3 of the definition of frontiers, we can conclude that $q' \subseteq_{\mathcal{O}} q$. Overall, we have shown $q' \equiv_{\mathcal{O}} q$. ❑

C Proofs for Section 4

Theorem 4. *AQ$^\wedge$s are not polynomial query learnable under disjointness ontologies using only membership queries.*

Proof. To prove the theorem, we use a proof strategy that is inspired by basic lower bound proofs for abstract learning problems due to Angluin [2]. Essentially the same proof is given in [15] for a slightly different class of ontologies that allows only concept inclusions between arbitrary conjunctions of concept names.

Here, it is convenient to view the oracle as an adversary who maintains a set S of candidate target queries that the learner cannot distinguish based on the queries made so far. We have to choose S and the ontology carefully so that each membership query removes only few candidate targets from S and that after a polynomial number of queries there is still more than one candidate that the learner cannot distinguish.

For each $n \geq 1$, let

$$\mathcal{O}_n = \{A_i \sqcap A'_i \sqsubseteq \bot \mid 1 \leq i \leq n\}$$

and

$$S_n = \{q(x) \leftarrow \alpha_1(x) \wedge \cdots \wedge \alpha_n(x) \mid \alpha_i \in \{A_i, A'_i\} \text{ for all } i \text{ with } 1 \leq i \leq n\}.$$

Note that S_n is a frontier of $\bot$ w.r.t. $\mathcal{O}_n$, if only AQ$^\wedge$ queries using the concept names A_i and A'_i for all $1 \leq i \leq n$, are considered for Condition 3. Clearly, S_n contains 2^n queries.[4]

[4] In fact, it can be shown similar as in the proof of Theorem 2 that S_n is contained in any frontier of $\bot$ w.r.t. $\mathcal{O}_n$. Hence, $\bot$ does not have polynomially sized frontiers w.r.t. disjointness ontologies.

Assume to the contrary of what is to be shown that $\mathrm{AQ}^\wedge$ queries are polynomial query learnable under disjointness ontologies using only membership queries. Then there exists a learning algorithm and polynomial p such that the number of membership queries needed to identify a target query q_T is bounded by $p(n_1, n_2)$, where n_1 is the size of q_T and n_2 is the size of the ontology. We choose n such that $2^n > p(r_1(n), r_2(n))$, where r_1 is a polynomial such that every query $q \in S_m$ satisfies $||q|| = r_1(m)$ and r_2 is a polynomial such that $r_2(m) > ||\mathcal{O}_m||$ for every $m \geq 1$.

Now, consider a membership query posed by the learning algorithm with ABox and answer individual $(\mathcal{A}, a)$. The oracle responds as follows:

1. if $\mathcal{A}, \mathcal{O}_n \models q(a)$ for no $q \in S_n$, then answer *no*;
2. if $\mathcal{A}, \mathcal{O}_n \models q(a)$ for a single $q \in S_n$, then answer *no* and remove q from S_n;
3. if $\mathcal{A}, \mathcal{O}_n \models q(a)$ for more than one $q \in S_n$, then answer *yes*.

Note that the third response is consistent since $\mathcal{A}$ must then contain $A_i(a)$ and $A'_i(a)$ for some i and thus $\mathcal{A}$ is not satisfiable w.r.t. $\mathcal{O}_n$. Moreover, the answers are always correct with respect to the updated set S_n. Thus, the learner cannot distinguish the remaining candidate queries by answers to queries posed so far.

It follows that the learning algorithm removes at most $p(r_1(n), r_2(n))$ many queries from S_n. By the choice of n, at least two candidate concepts remain in S_n after the algorithm is finished. Thus the learner cannot distinguish between them and we have derived a contradiction. ❑

Lemma 5. *If ELIQs are polynomial time learnable under DL-Lite ontologies in normal form using membership queries and a single initial equivalence query, then this is also true for unrestricted DL-Lite ontologies. The same holds for DL-Lite$^-$ ontologies without the initial equivalence query.*

Proof. We show the lemma by converting a learning algorithm L' for ontologies in normal form into a learning algorithm L for unrestricted ontologies. Since L will ask a single query for every query asked by L', the lemma follows.

Given an *DL-Lite* or *DL-Lite*$^-$ ontology $\mathcal{O}$ and a signature $\Sigma = \mathsf{sig}(\mathcal{O}) \cup \mathsf{sig}(q_T)$, algorithm L first computes the ontology $\mathcal{O}'$ in normal form as per Lemma 15, choosing the fresh concept names so that they are not from Σ. It then runs L' on $\mathcal{O}'$ and $\Sigma' = \Sigma \cup \mathsf{sig}(\mathcal{O}')$. In contrast to L', the oracle still works with the original ontology $\mathcal{O}$. To ensure that the answers to the queries posed to the oracle are correct, L modifies L' as follows.

Whenever L' asks a membership query $\mathcal{A}', \mathcal{O}' \models q_T(a)$, L instead asks the membership query $\mathcal{A}, \mathcal{O} \models q_T(a)$, where $\mathcal{A}$ is obtained from $\mathcal{A}'$ by replacing each concept assertion $X_{\exists R}(b)$ where $X_{\exists R}$ is a fresh concept name added during conversion to normal form with a role assertion $R(b, b')$ where b' is a fresh individual name. By the following claim, the answer to the modified membership query coincides with that to the original query.

Claim 1. $\mathcal{A}', \mathcal{O}' \models q(a)$ iff $\mathcal{A}, \mathcal{O} \models q(a)$ for all ELIQs q that only use symbols from Σ.

Proof of Claim 1. $\mathcal{A}'$ is not satisfiable w.r.t. $\mathcal{O}'$ iff $\mathcal{A}$ is not satisfiable w.r.t. $\mathcal{O}$, since the conversion to normal form does not add any disjointness constraints. Hence it remains to show the claim for satisfiable ABoxes. For "if", suppose that $\mathcal{A}, \mathcal{O} \models q(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}'$ and $\mathcal{O}'$. Since $\mathcal{O}'$ is a conservative extension of $\mathcal{O}$, $\mathcal{I}$ is also a model of $\mathcal{O}$. Additionally, $\mathcal{I}$ satisfies $X_{\exists R} \sqsubseteq \exists R$ for all concept names $X_{\exists R}$ that are added during conversion of $\mathcal{O}$ to normal form. Thus $\mathcal{I}$ is also a model of $\mathcal{A}$ and $\mathcal{I} \models q(a)$. For "only if", suppose that $\mathcal{A}', \mathcal{O}' \models q(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}$ and $\mathcal{O}$. Since $\mathcal{O}'$ is a conservative extension of $\mathcal{O}$, there is a model $\mathcal{I}'$ of $\mathcal{O}'$ that coincides with $\mathcal{I}$ on all symbols from Σ. $\mathcal{I}'$ must satisfy $\exists R \sqsubseteq X_{\exists R}$ for all fresh concept names $X_{\exists R}$. Thus $\mathcal{I}'$ is a model of $\mathcal{A}'$. Since $\mathsf{sig}(q) \subseteq \Sigma$ and $\mathcal{I}'$ and $\mathcal{I}$ coincide on Σ, it follows that $\mathcal{I} \models q(a)$ as required.

Second, whenever L' asks an equivalence query $q'_H \equiv_{\mathcal{O}'} q_T$, L instead asks the equivalence query $q_H \equiv_{\mathcal{O}} q_T$, where q_H is obtained from q'_H by replacing each atom $X_{\exists R}(y)$ with the atom $R(y, y')$ where y' is a fresh variable. Applying the following claim to both q'_H and $q'_T = q_T$, the answer to the modified equivalence query coincides with that to the original query.

Claim 2. Let q' be an ELIQ that uses only symbols from Σ' and let q be obtained from q' by replacing each atom $R_{\exists R}(y)$ with the atom $R(y, y')$, where y' is a fresh variable. Then $\mathcal{A}, \mathcal{O}' \models q'(a)$ iff $\mathcal{A}, \mathcal{O} \models q(a)$ for all ABoxes $\mathcal{A}$ using only symbols from Σ.

Proof of Claim 2. Again, the claim holds trivially for not satisfiable ABoxes. Therefore assume $\mathcal{A}$ to be satisfiable. For "if", suppose $\mathcal{A}, \mathcal{O} \models q(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}$ and $\mathcal{O}'$. Since $\mathcal{O}'$ is a conservative extension, $\mathcal{I}$ is also a model of $\mathcal{O}$, thus $\mathcal{I} \models q(a)$. Since $\mathcal{I}$ also must satisfy $\exists R \sqsubseteq X_{\exists R}$ for all fresh concept names $X_{\exists R}$, $\mathcal{I} \models q'(a)$.

For "only if", suppose $\mathcal{A}, \mathcal{O}' \models q'(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}$ and $\mathcal{O}$. Since $\mathcal{O}'$ is a conservative extension of $\mathcal{O}$, there is a model $\mathcal{I}'$ of $\mathcal{O}'$ that coincides on all symbols from Σ with $\mathcal{I}$ and is also a model of $\mathcal{A}$. Thus $\mathcal{I}' \models q'(a)$. Since $\mathcal{I}'$ must satisfy $X_{\exists R} \sqsubseteq \exists R$ for all $X_{\exists R}$, $\mathcal{I} \models q(a)$ as required. ❑

Lemma 6. *Let q be a unary CQ that is $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$ such that $q \subseteq q_T$ for the target query $q_T(y)$, and let $q' = \mathsf{minimize}(q)$. Then*

1. *$q \subseteq_{\mathcal{O}} q'$ and $q' \subseteq_{\mathcal{O}} q_T$;*
2. *$\mathsf{var}(q') \subseteq \mathsf{img}(h)$ for every homomorphism h from q_T to $\mathcal{U}_{q',\mathcal{O}}$ with $h(y) = x$ (and consequently $|\mathsf{var}(q')| \leq |\mathsf{var}(q_T)|$);*
3. *q' is $\mathcal{O}$-minimal.*

Proof. Point 1 of the Lemma holds since q' is a subset of q and $\mathsf{minimize}$ ensures in each step that $\mathcal{A}_{q'}, \mathcal{O} \models q_T(x)$, which implies $q' \subseteq_{\mathcal{O}} q_T$.

For Point 2, let y be the answer variable of q_T and assume that there is a homomorphism h from q_T to $\mathcal{U}_{q',\mathcal{O}}$ with $h(y) = x$ and a $x' \in \mathsf{var}(q')$ that is not in $\mathsf{img}(h)$. Let q'' be the result of removing from q' all atoms that involve x'. We show that h is a homomorphism from q_T to $\mathcal{U}_{q'',\mathcal{O}}$ which witnesses that $\mathcal{A}_{q''}, \mathcal{O} \models q_T(x)$. Hence all role atoms involving x' are dropped by $\mathsf{minimize}$, in contradiction to $x' \in \mathsf{var}(q')$. To see that h is indeed a homomorphism, first note that for all $x_1, x_2 \in \mathsf{var}(q') \setminus \{x'\}$, the following holds by $\mathcal{O}$-saturation of q and construction of universal models:

1. $A(x_1) \in \mathcal{U}_{q',\mathcal{O}}$ iff $A(x_1) \in \mathcal{U}_{q'',\mathcal{O}}$;
2. $r(x_1, x_2) \in \mathcal{U}_{q',\mathcal{O}}$ iff $r(x_1, x_2) \in \mathcal{U}_{q'',\mathcal{O}}$.

From the normal form of $\mathcal{O}$ and 1., it follows that the subtree in $\mathcal{U}_{q',\mathcal{O}}$ below each $x_1 \in \mathsf{var}(q') \setminus \{x'\}$ is identical to the subtree in $\mathcal{U}_{q'',\mathcal{O}}$ below x_1. Thus h is a homomorphism.

For Point 3, assume that q' is not minimal, that is, there exists a homomorphism h from q' to $\mathcal{U}_{q'_S,\mathcal{O}}$ with $h(x) = x$ where q_S is a restriction of q' to a proper subset of $\mathsf{var}(q')$. Then h is also a homomorphism from q' to $\mathcal{U}_{q',\mathcal{O}}$ with $\mathsf{var}(q') \not\subseteq \mathsf{img}(h)$. By Lemma 13, h can be extended to a homomorphism h' from $\mathcal{U}_{q',\mathcal{O}}$ to $\mathcal{U}_{q',\mathcal{O}}$ without adding any element of $\mathsf{var}(q')$ to $\mathsf{img}(h')$. Composing a homomorphism from q_T to $\mathcal{U}_{q',\mathcal{O}}$ that exists by Point 1 with h' yields a homomorphism g from q_T to $\mathcal{U}_{q',\mathcal{O}}$ with $\mathsf{var}(q') \not\subseteq \mathsf{img}(g)$ and $g(y) = x$, contradicting Point 2. ❑

Lemma 7. *For all $i \geq 1$,*

1. *p_i is $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$;*
2. *$p_i \subseteq_{\mathcal{O}} q_T$;*
3. *$p_i \subseteq_{\mathcal{O}} p_{i+1}$ and $|\mathsf{var}(p_{i+1})| > |\mathsf{var}(p_i)|$.*

Proof. For Point 1 of the Lemma, observe that both doubling the length of a cycle and $\mathsf{minimize}$, preserve $\mathcal{O}$-saturation and satisfiability w.r.t. $\mathcal{O}$. Since $p_1 = \mathsf{minimize}(q)$ is $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$, so is every p_i.

We show Point 2 by induction on i. The case $i = 1$ is immediate since $p_1 = \mathsf{minimize}(q)$ and $q \subseteq_{\mathcal{O}} q_T$. Now let $i \geq 1$. By the induction hypothesis $p_i \subseteq q_T$ and thus $\mathcal{A}_{p_i}, \mathcal{O} \models q_T(x)$. Assume that p'_i was obtained from p_i by expanding a cycle $R_1(x_1, x_2), \ldots, R_n(x_n, x_1)$. Then

$$S = \{(y, y) \mid y \in \mathsf{var}(p_i)\} \cup \{(x_i, x'_i) \mid 1 \leq i \leq n\}$$

is an $\mathcal{ELI}$-simulation from $\mathcal{A}_{p_i}$ to $\mathcal{A}_{p'_i}$ with $(x, x) \in S$. Therefore by Lemma 14 $\mathcal{A}_{p'_i}, \mathcal{O} \models q_T(x)$ and $p'_i \subseteq_{\mathcal{O}} q_T$. Hence, by Point 1 of Lemma 6, $p_{i+1} \subseteq_{\mathcal{O}} q_T$ for $p_{i+1} = \mathsf{minimize}(p'_i)$.

For Point 3, define a mapping g from $\mathsf{var}(p_{i+1})$ to $\mathsf{var}(p_i)$ by setting $g(y) = y$ for all $y \in \mathsf{var}(p_i) \cap \mathsf{var}(p_{i+1})$ and $g(y') = y$ for all $y' \in \mathsf{var}(p_{i+1}) \setminus \mathsf{var}(p_i)$. We show that g is a homomorphism from p_i to $\mathcal{U}_{p_{i+1},\mathcal{O}}$, implying $p_i \subseteq_{\mathcal{O}} p_{i+1}$. If $A(y) \in p_{i+1}$, then $A(y) \in p'_i$ by the definition of $\mathsf{minimize}$ and $A(g(y)) \in p_i$ by the construction of p'_i. If $r(y, y') \in p_{i+1}$, then $r(y, y') \in p'_i$ by the definition of $\mathsf{minimize}$ and $r(g(y), g(y')) \in p_i$ by the construction of p_i.

For $|\mathsf{var}(p_{i+1})| > |\mathsf{var}(p_i)|$, we will show the following three claims, proving that g is surjective, but not injective. For an injective and surjective function, we use g^- to denote the inverse of g. Again, let y be the answer variable of p_T.

Claim 1. g is surjective.

Proof of Claim 1. Suppose that g is not surjective. Then $\mathsf{var}(p_i) \not\subset \mathsf{img}(g)$. Recall that $p_i = \mathsf{minimize}(q)$ for some query q and thus by Point 3 of Lemma 6, $\mathsf{var}(q') \subseteq \mathsf{img}(h)$ for every homomorphism h from q_T to $\mathcal{U}_{p_i,\mathcal{O}}$ with $h(y) = x$.

Let h_2 be the extension of g to a homomorphism from $\mathcal{U}_{p_{i+1},\mathcal{O}}$ to $\mathcal{U}_{p_i,\mathcal{O}}$ as in Lemma 13. Note that $\mathsf{var}(p_i) \not\subseteq \mathsf{img}(h_2)$. By Point 2, there is a homomorphism h_1 from q_T to $\mathcal{U}_{p_{i+1},\mathcal{O}}$ with $h(y) = x$. Composing h_1 and h_2 yields a homomorphism h_3 from q_T to $\mathcal{U}_{p_i,\mathcal{O}}$ with $h_3(y) = x$ and $\mathsf{var}(p_i) \not\subseteq \mathsf{img}(h_3)$, a contradiction.

Claim 2. If g is injective, then $r(y_1, y_2) \in p_i$ implies $r(g^-(y_1), g^-(y_2)) \in p_{i+1}$.

Proof of Claim 2. Suppose to the contrary that there is an $r(y_1, y_2) \in p_i$ with $r(g^-(y_1), g^-(y_2)) \notin p_{i+1}$. It is then also a homomorphism from p_{i+1} to $p_i \setminus \{r(y_1, y_2)\}$ and using the same composition-of-homomorphisms argument as in the proof of Claim 1, we find a homomorphism h from q_T to $\mathcal{U}_{p_i \setminus \{r(y_1,y_2)\},\mathcal{O}}$ with $h(y) = x$. Hence $p_i \setminus \{r(y_1, y_2)\} \subseteq_\mathcal{O} q_T$. This contradicts the fact that $p_i = \mathsf{minimize}(q)$ for some query q.

Claim 3. g is not injective.

Proof of Claim 3. Let $R_1(x_1, x_2), \ldots, R_n(x_n, x_1) \in p_i$ be the cycle that is expanded during the construction of p_{i+1} from p_i. Without loss of generality, assume that $R_n = r_n$ is a role name, but not an inverse role. Suppose for contradiction that g is injective. The construction of g, together with g being surjective and injective, implies that exactly one of x_j, x'_j is in $\mathsf{var}(p_{i+1})$ for all j with $1 \leq j \leq n$. Assume that $x_n \in \mathsf{var}(p_{i+1})$ (the case $x'_n \in \mathsf{var}(p_{i+1})$ is analogous) and thus $g(x_n) = x_n$.

We prove by induction on j that $x_j \notin \mathsf{var}(p_{i+1})$ for $1 \leq j \leq n$, thus obtaining a contradiction to $x_n \in \mathsf{var}(p_{i+1})$. For the induction start, assume to the contrary of what is to be shown that $x_1 \in \mathsf{var}(p_{i+1})$. Then $g(x_1) = x_1$ and $r_n(x_n, x_1) \in p_i$ implies $r_n(x_n, x_1) \in p_{i+1}$ by Claim 2. A contradiction to the construction of p_{i+1}.

For the induction step, let $j \geq 1$. By the induction hypothesis $x_j \notin \mathsf{var}(p_{i+1})$ and thus $x'_j \in \mathsf{var}(p_{i+1})$. Then $g(x'_j) = x_j$. Assume to the contrary of what is shown that $x_{j+1} \in \mathsf{var}(p_{i+1})$. Then $g(x_{j+1}) = x_{j+1}$ and $R_j(x_j, x_{j+1}) \in p_i$ yield $R_j(x'_j, x_{j+1}) \in p_{i+1}$ by Claim 2. A contradiction to the construction of p_{i+1}. ❑

Lemma 8. *Let q be a CQ that is $\mathcal{O}$-saturated, satisfiable w.r.t. $\mathcal{O}$, and satisfies $q \subseteq_\mathcal{O} q_T$. Then $q' = \mathsf{treeify}(q)$ is an ELIQ that can be computed in time polynomial in $|\mathsf{var}(q_T)| + ||q||$ using membership queries.*

Proof. Let $p_1, p_2, \ldots,$ be the sequence of constructed queries. Recall that for all $i \geq 1$, p_i is the result of applying $\mathsf{minimize}$ to some query. Thus by Lemma 6 Point 2, $|\mathsf{var}(p_i)| \leq |\mathsf{var}(q_T)|$ for all $i \geq i$. But by Lemma 7 Point 3, the number of variables in p_i increases with every doubling of the length of some cycle. The length n of the sequence of queries is therefore at most $|\mathsf{var}(q_T)|$ and $\mathsf{treeify}$ stops at $p_n = \mathsf{treeify}(q)$. Hence p_n does not contain a cycle, making it an ELIQ.

It remains to show that every operation runs in polynomial time. Clearly, $|\Sigma| \leq ||q||$ and since the operation does not introduce new concept or role names, $\mathsf{sig}(p_i) \subseteq \Sigma$ for all i.

Each call to $\mathsf{minimize}$ makes at most $|\Sigma| \cdot |\mathsf{var}(p_i)|^2$ membership queries. For the operation of $\mathsf{treeify}$, note that a cycle can be identified in time polynomial in $|\mathsf{var}(p_i)| \leq |\mathsf{var}(q_T)|$. ❑

Lemma 9. *For all $i \geq 1$,*

1. $q_i \subseteq_{\mathcal{O}} q_T$*;*
2. $q_i \subseteq_{\mathcal{O}} q_{i+1}$ *and* $q_{i+1} \not\subseteq_{\mathcal{O}} q_i$*;*
3. $\mathsf{var}(q_i) \subseteq \mathsf{img}(h)$ *for every homomorphism h from q_{i+1} to $\mathcal{U}_{q_i,\mathcal{O}}$ with $h(x) = x$.*

Proof. For Point 1, first consider the case $i = 1$. Then $q_1 = \mathsf{treeify}(q_H^0)$, and by $q_H^0 \subseteq_{\mathcal{O}} q_T$ and Lemma 7 Point 1 it follows that $q_1 \subseteq_{\mathcal{O}} q_T$. For $i > 1$, recall that $q_i = \mathsf{minimize}(q_F)$ for some $q_F \subseteq_{\mathcal{O}} q_T$. Thus $q_i \subseteq_{\mathcal{O}} q_T$ follows by Lemma 6 Point 1.

Point 2 follows from the definition of a frontier of q_i w.r.t $\mathcal{O}$ and the property of $\mathsf{minimize}$ in Lemma 6 Point 1.

For Point 3, let h be a homomorphism from q_{i+1} to $\mathcal{U}_{q_i,\mathcal{O}}$ with $h(x) = x$. By Lemma 13, we can extend h to a homomorphism h' from $\mathcal{U}_{q_{i+1},\mathcal{O}}$ to $\mathcal{U}_{q_i,\mathcal{O}}$ such that no elements of $\mathsf{var}(q_i)$ are added to $\mathsf{img}(h')$. By Point 1, there is a homomorphism g from q_T to $\mathcal{U}_{q_{i+1},\mathcal{O}}$ with $h(y) = x$. Composing g and h' yields a homomorphism g' from q_T to $\mathcal{U}_{q_i,\mathcal{O}}$ with $g'(y) = x$. Recall that for all $i \geq 1$, $q_i = \mathsf{minimize}(q)$ for some ELIQ q that is $\mathcal{O}$-saturated and satisfiable w.r.t $\mathcal{O}$, thus $\mathsf{var}(q_i) \subseteq \mathsf{img}(g')$ by Lemma 6 Point 2. Since $\mathsf{img}(g') \subseteq \mathsf{img}(h')$, it follows that $\mathsf{var}(q_i) \subseteq \mathsf{img}(h)$, as required. ❑

Lemma 10. *$q_n \equiv q_T$ for some $n \leq p(|\mathsf{var}(q_T)| + |\Sigma|)$, p a polynomial.*

Proof. For all $i \geq 1$, $q_i = \mathsf{minimize}(q)$ for some ELIQ q that is $\mathcal{O}$-saturated and satisfiable w.r.t $\mathcal{O}$, thus by Lemma 6 Point 2, $|\mathsf{var}(q_i)| \leq |\mathsf{var}(q_T)|$. Moreover, Lemma 9 Point 3 implies that $|\mathsf{var}(q_i)| \leq |\mathsf{var}(q_{i+1})|$. Hence, it remains to show that the length of any subsequence $q_j, \dots, q_k$ with $|\mathsf{var}(q_j)| = \cdots = |\mathsf{var}(q_k)|$ is bounded by a polynomial in $\mathsf{var}(q_T)$ and $|\Sigma|$.

Let h_ℓ for $\ell \in \{j+1, \dots, k\}$ be the homomorphism from q_ℓ to $\mathcal{U}_{q_{\ell+1},\mathcal{O}}$ that exists due to Lemma 9 Point 2. Since $|\mathsf{var}(q_\ell)| = |\mathsf{var}(q_{\ell+1})|$, h_ℓ is a bijection between $\mathsf{var}(q_{\ell+1})$ and $\mathsf{var}(q_\ell)$. By Lemma 9 Point 2, h^- is not a homomorphism from q_ℓ to $q_{\ell+1}$.

Therefore there is either a concept atom $A(x_1) \in q_\ell$ such that $A(h_\ell^-(x_1)) \notin q_{\ell+1}$ or there is a role atom $r(x_1, x_2) \in q_\ell$ such that $r(h_\ell^-(x_1), h_\ell^-(x_2)) \notin q_{\ell+1}$. The second case is not possible, since h^- is a bijection and both q_ℓ and $q_{\ell+1}$ are ELIQs. Thus q_ℓ contains at least one concept atom more than $q_{\ell+1}$ and the length of the sequence $q_j, \dots, q_k$ is bounded by $|\mathsf{var}(q_T)| \cdot |\Sigma|$. ❑