Frontiers and Exact Learning of $\mathcal{ELI}$ Queries under DL-Lite Ontologies

Maurice Funk[1], **Jean Christoph Jung**[2] and **Carsten Lutz**[1]
[1] Leipzig University, Faculty of Mathematics and Computer Science, Germany
[2] University of Hildesheim, Institute of Computer Science, Germany
mfunk@informatk.uni-leipzig.de, jungj@uni-hildesheim.de, clu@informatik.uni-leipzig.de

Abstract

We study $\mathcal{ELI}$ queries (ELIQs) in the presence of ontologies formulated in the description logic *DL-Lite*. For the dialect *DL-Lite*$^\mathcal{H}$, we show that ELIQs have a frontier (set of least general generalizations) that is of polynomial size and can be computed in polynomial time. In the dialect *DL-Lite*$^\mathcal{F}$, in contrast, frontiers may be infinite. We identify a natural syntactic restriction that enables the same positive results as for *DL-Lite*$^\mathcal{H}$. We use our results on frontiers to show that ELIQs are learnable in polynomial time in the presence of a *DL-Lite*$^\mathcal{H}$/ restricted *DL-Lite*$^\mathcal{F}$ ontology in Angluin's framework of exact learning with only membership queries.

1 Introduction

In the widely studied paradigm of ontology-mediated querying, a database query is enriched with an ontology that provides domain knowledge as well as additional vocabulary for query formulation [Bienvenu *et al.*, 2013b; Calvanese *et al.*, 2009]. We consider ontologies formulated in description logics (DLs) of the *DL-Lite* family and queries that are $\mathcal{ELI}$ queries (ELIQs) or, in other words, tree-shaped unary conjunctive queries (CQs). *DL-Lite* is a prominent choice for the ontology language as it underpins the OWL 2 QL profile of the OWL ontology language [OWL Working Group, 2009]. Likewise, ELIQs are a prominent choice for the query language as they are computationally very well-behaved: without an ontology, they can be evaluated in polynomial time in combined complexity, in contrast to NP-completeness for unrestricted CQs. Moreover, in the form of $\mathcal{ELI}$ concepts they are a central building block of ontologies in several dialects of *DL-Lite* and beyond.

The aim of this paper is to study the related topics of computing least general generalizations (LGGs) of ELIQs under *DL-Lite* ontologies and learning ELIQs under *DL-Lite* ontologies in Angluin's framework of exact learning [Angluin, 1987a; Angluin, 1987b]. Computing generalizations is a natural operation in query engineering that plays a crucial role in learning logical formulas [Plotkin, 1970; Muggleton, 1991], in particular in exact learning [ten Cate and Dalmau, 2021]. Exact learning, in turn, is concerned with constructing queries and ontologies. This can be challenging and costly, especially when logic expertise and domain knowledge are not in the same hands. Aiming at such cases, exact learning provides a systematic protocol for query engineering in which a learner interacts in a game-like fashion with an oracle, which may be a domain expert.

Our results on LGGs concern the notion of a *frontier* of an ELIQ q w.r.t. an ontology $\mathcal{O}$. Such a frontier is a set $\mathcal{F}$ of ELIQs that *generalize* q, that is, $q \subseteq_\mathcal{O} q_F$ and $q_F \not\subseteq_\mathcal{O} q$ for all $q_F \in \mathcal{F}$, where '$\subseteq_\mathcal{O}$' denotes query containment w.r.t. $\mathcal{O}$. Moreover, $\mathcal{F}$ must be *complete* in that 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 $q_F \subseteq_\mathcal{O} q'$. We are interested in computing a frontier that contains only polynomially many ELIQs of polynomial size, in polynomial time. This is possible in the case of ELIQs without ontologies as shown in [ten Cate and Dalmau, 2021]; for the simpler $\mathcal{EL}$ queries, the same had been observed earlier (also without ontologies) in [Baader *et al.*, 2018; Kriegel, 2019]. In contrast, unrestricted CQs do not even admit finite frontiers [Nesetril and Tardif, 2000].

In exact learning, the learner and the oracle know and agree on the ontology $\mathcal{O}$, and they also agree on the target query q_T to use only concept and role names from $\mathcal{O}$. The learner may ask *membership queries* where they produce an ABox $\mathcal{A}$ and a candidate answer a and ask whether $\mathcal{A}, \mathcal{O} \models q_T(a)$, that is, whether a is an answer to q_T w.r.t. $\mathcal{O}$ on $\mathcal{A}$. The oracle faithfully answers "yes" or "no". *Polynomial time learnability* then means that the learner has an algorithm for constructing q_T, up to equivalence w.r.t. $\mathcal{O}$, with running time bounded by a polynomial in the sizes of q_T and $\mathcal{O}$.

Learning with only membership queries, as described above and studied in this article, is a strong form of exact learning. In fact, there are not many cases where polynomial time learning with only membership queries is possible, ELIQs without ontologies being an important example [ten Cate and Dalmau, 2021]. Often, one would therefore also admit *equivalence queries* where 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 answer a such that $\mathcal{A}, \mathcal{O} \models q_T(a)$ and $\mathcal{A}, \mathcal{O} \not\models q_H(a)$ or vice versa. This is done, for instance, in [Konev *et al.*, 2018; Funk *et al.*, 2021].

We consider as ontology languages the DLs *DL-Lite*$^\mathcal{H}$ and

DL-Lite$^{\mathcal{F}}$, equipped with *role inclusions* (also known as role hierarchies) and *functional roles*, respectively. Both dialects admit concept and role disjointness constraints and $\mathcal{ELI}$ concepts on the right-hand side of concept inclusions [Calvanese *et al.*, 2007; Kikot *et al.*, 2011]. We show that *DL-Lite*$^{\mathcal{H}}$ admits polynomial frontiers that can be computed in polynomial time, and that *DL-Lite*$^{\mathcal{F}}$ does not even admit finite frontiers. We then introduce a fragment *DL-Lite*$^{\mathcal{F}-}$ of *DL-Lite*$^{\mathcal{F}}$ that restricts the use of inverse functional roles on the right-hand side of concept inclusions and show that it is as well-behaved as *DL-Lite*$^{\mathcal{H}}$. Both frontier constructions require a rather subtle analysis. We also note that adding conjunction results in frontiers of exponential size, even for very simple fragments of *DL-Lite*. One application of our results is to show that every ELIQ q can be characterized up to equivalence w.r.t. ontologies formulated in *DL-Lite*$^{\mathcal{H}}$ or *DL-Lite*$^{\mathcal{F}-}$ by only polynomially many data examples of the form $(\mathcal{A}, a)$, labeled as positive if $\mathcal{A}, \mathcal{O} \models q(a)$ and as negative otherwise.

We then consider in detail the application of our results in the context of exact learning and show that ELIQs can be learned in polynomial time w.r.t. ontologies $\mathcal{O}$ formulated in *DL-Lite*$^{\mathcal{H}}$ or *DL-Lite*$^{\mathcal{F}-}$. The learning algorithm uses only membership queries provided that a seed query is available, that is, an ELIQ q_0 such that $q_0 \subseteq_{\mathcal{O}} q_T$. Such a seed query can be constructed using membership queries if $\mathcal{O}$ contains no concept disjointness constraints and obtained by a single initial equivalence query otherwise. We also show that ELIQs cannot be learned at all w.r.t. unrestricted *DL-Lite*$^{\mathcal{F}}$ ontologies using only membership queries, and that they cannot be learned with only polynomially many membership queries when conjunction is admitted.

Proof details are in the appendix.

Related Work. Exact learning of queries in the context of description logics has been studied in [Funk *et al.*, 2021] while [Konev *et al.*, 2018] considers learning entire ontologies, see also [Ozaki *et al.*, 2020; Ozaki, 2020]. It is shown in [Funk *et al.*, 2021] that a restricted form of CQs (that do not encompass all ELIQs) can be learned in polynomial time under $\mathcal{EL}$ ontologies using both membership and equivalence queries. The results from that paper indicate that inverse roles provide a challenge for exact learning under ontologies and thus it is remarkable that we can handle them without any restrictions in our context. Related forms of learning are the construction of the least common subsumer (LCS) and the most specific concept (MSC) [Baader, 2003; Baader *et al.*, 1999; Baader *et al.*, 2007; Jung *et al.*, 2020b; Zarrieß and Turhan, 2013] which may both be viewed as a form of query generalization. There is also a more loosely related research thread on learning DL concepts from labeled data examples [Funk *et al.*, 2019; Jung *et al.*, 2020a; Lehmann and Hitzler, 2010; Lehmann and Völker, 2014; Sarker and Hitzler, 2019].

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 $r \in \mathsf{N_R}$ or the inverse r^- of a role name r. An $\mathcal{ELI}$ *concept* is formed according to the syntax rule $C, D ::= \top \mid A \mid C \sqcap D \mid \exists R.C$ where A ranges over concept names and R over roles. A *basic concept* B is an $\mathcal{ELI}$ concept of the form $\top$, A, or $\exists R.\top$. When dealing with basic concepts, for brevity we may write $\exists R$ in place of $\exists R.\top$.

A *DL-Lite*$^{\mathcal{HF}}$ ontology $\mathcal{O}$ is a finite set of *concept inclusions (CIs)* $B \sqsubseteq C$, *role inclusions (RIs)* $R_1 \sqsubseteq R_2$, *concept disjointness constraints* $B_1 \sqcap B_2 \sqsubseteq \bot$, *role disjointness constraints* $R_1 \sqcap R_2 \sqsubseteq \bot$, and *functionality assertions* $\mathsf{func}(R)$. Here, B, B_1, and B_2 range over basic concepts, C over $\mathcal{ELI}$ concepts, and R_1, R_2, R over roles. Superscript $\cdot^{\mathcal{H}}$ indicates the presence of role inclusions (also called role hierarchies) and superscript $\cdot^{\mathcal{F}}$ indicates functionality assertions, and thus it should be clear what we mean with a *DL-Lite*$^{\mathcal{H}}$ ontology and with a *DL-Lite*$^{\mathcal{F}}$ ontology. In fact, we are mainly interested in these two fragments of *DL-Lite*$^{\mathcal{HF}}$.

A *DL-Lite*$^{\mathcal{HF}}$ ontology is in *normal form* if all concept inclusions in it are of one of the forms $A \sqsubseteq B$, $B \sqsubseteq A$, and $A \sqsubseteq \exists R.A'$ with A, A' concept names or $\top$ and B a basic concept. Note that CIs of the form $\exists R \sqsubseteq \exists S$ are not admitted and neither are CIs of the form $A \sqsubseteq \exists R.C$ with C a compound 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 $\mathcal{A}$.

As usual, the semantics is given in terms of *interpretations* $\mathcal{I}$, which we define to be a (possibly infinite and) nonempty 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 $r \in \mathsf{N_R}$. This definition of interpretation is slightly different from the usual one, but equivalent;[1] its virtue is uniformity as every ABox is a finite interpretation. The interpretation function $\cdot^{\mathcal{I}}$ can be extended from concept names to $\mathcal{ELI}$ concepts in the standard way [Baader *et al.*, 2017]. An interpretation $\mathcal{I}$ *satisfies* a concept or role inclusion $\alpha_1 \sqsubseteq \alpha_2$ if $\alpha_1^{\mathcal{I}} \subseteq \alpha_2^{\mathcal{I}}$, a concept or role disjointness constraint $\alpha_1 \sqcap \alpha_2 \sqsubseteq \bot$ if $\alpha_1^{\mathcal{I}} \cap \alpha_2^{\mathcal{I}} = \emptyset$, and a functionality assertion $\mathsf{func}(R)$ if $R^{\mathcal{I}}$ is a partial function. It *satisfies* a concept or role assertion α if $\alpha \in \mathcal{I}$. Note that, as usual, we thus make the standard names assumption, implying the unique name assumption.

An interpretation is a *model* of an ontology or an ABox if it satisfies all inclusions, disjointness constraints, and assertions in it. We write $\mathcal{O} \models \alpha_1 \sqsubseteq \alpha_2$ if every model of the ontology $\mathcal{O}$ satisfies the concept or role inclusion $\alpha_1 \sqsubseteq \alpha_2$ and $\mathcal{O} \models \alpha_1 \equiv \alpha_2$ if $\mathcal{O} \models \alpha_1 \sqsubseteq \alpha_2$ and $\mathcal{O} \models \alpha_2 \sqsubseteq \alpha_1$. If α_1 and α_2 are basic concepts or roles, then such consequences are decidable in PTIME both in *DL-Lite*$^{\mathcal{H}}$ and in *DL-Lite*$^{\mathcal{F}}$ [Artale *et al.*, 2009]. An ABox $\mathcal{A}$ is *satisfiable* w.r.t. an ontology $\mathcal{O}$ if $\mathcal{A}$ and $\mathcal{O}$ have a common model. Deciding ABox satisfiability is also in PTIME in both *DL-Lite*$^{\mathcal{H}}$ and *DL-Lite*$^{\mathcal{F}}$.

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

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

of a representation of O as a word in a suitable alphabet.

Queries. An $\mathcal{ELI}$ concept C can be viewed as an $\mathcal{ELI}$ *query (ELIQ)*. An individual $a \in \mathsf{ind}(\mathcal{A})$ is an *answer* to C on an ABox $\mathcal{A}$ w.r.t. an ontology $\mathcal{O}$, written $\mathcal{A}, \mathcal{O} \models C(a)$, if $a \in C^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{O}$. We shall often view ELIQs as unary *conjunctive queries (CQs)* and also consider CQs that are not ELIQs. In this paper, CQs are always unary. A CQ thus takes the form $q(x_0) = \exists \bar{y}\, \phi(x_0, \bar{y})$ with ϕ a conjunction of *concept atoms* $A(x)$ and *role atoms* $r(x, y)$ where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. We use $\mathsf{var}(q)$ to denote the set of variables that occur in q. We may view q as a set of atoms and may write $r^-(x, y)$ in place of $r(y, x)$. We call x_0 the *answer variable* and use the notion of an answer and the notation $\mathcal{A}, \mathcal{O} \models q(a)$ also for CQs. The formal definition is in terms of homomorphisms as usual, details are in the appendix. ELIQs are in 1-to-1 correspondence with CQs whose Gaifman graph is a tree and that contain no self-loops and multi-edges. For example, the ELIQ $C = A \sqcap \exists r^-.(\exists s.B \sqcap \exists r.A)$ is the CQ $q(x_0) = \{A(x), r(y, x), s(y, z), B(z), r(y, z'), A(z')\}$. We use $\mathcal{A}_q$ to denote the ABox obtained from CQ q by viewing variables as individuals and atoms as assertions. A CQ q is *satisfiable* w.r.t. ontology $\mathcal{O}$ if $\mathcal{A}_q$ is.

For CQs q_1 and q_2 and an ontology $\mathcal{O}$, 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 $\mathcal{A}, \mathcal{O} \models q_2(a)$. If q_1, q_2 are ELIQs, then this coincides with q_1 viewed as $\mathcal{ELI}$ concept being subsumed w.r.t. $\mathcal{O}$ by q_2 viewed as an $\mathcal{ELI}$ concept [Baader *et al.*, 2017]. 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. A CQ q is *$\mathcal{O}$-saturated*, with $\mathcal{O}$ an 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 $x \in \mathsf{var}(q)$ such that $q \equiv_{\mathcal{O}} q|_{\mathsf{var}(q)\setminus\{x\}}$ with $q|_S$ the restriction of q to the atoms that only contain variables in S. For a CQ q and an ontology $\mathcal{O}$ formulated in *DL-Lite*$^{\mathcal{H}}$ or *DL-Lite*$^{\mathcal{F}}$, one can easily find in polynomial time an $\mathcal{O}$-saturated CQ q' with $q \equiv_{\mathcal{O}} q'$. To achieve $\mathcal{O}$-minimality, we may repeatedly choose variables $x \in \mathsf{var}(q)$, check whether $\mathcal{A}_{q|_{\mathsf{var}(q)\setminus\{x\}}}, \mathcal{O} \models q$, and if so replace q with $q|_{\mathsf{var}(q)\setminus\{x\}}$. For ELIQs, the required checks can be carried out in PTIME in *DL-Lite*$^{\mathcal{F}}$ [Bienvenu *et al.*, 2013a], but are NP-complete in *DL-Lite*$^{\mathcal{H}}$ [Kikot *et al.*, 2011]. We conjecture that in *DL-Lite*$^{\mathcal{H}}$, it is not possible to construct equivalent $\mathcal{O}$-minimal ELIQs in polynomial time.

3 Frontiers in *DL-Lite*$^{\mathcal{H}}$

We show that for every ELIQ q and *DL-Lite*$^{\mathcal{H}}$ 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. We also observe that this fails when *DL-Lite*$^{\mathcal{H}}$ is extended with conjunction, even in very restricted cases.

Definition 1. *A* frontier *of an ELIQ q w.r.t. a DL-Lite$^{\mathcal{HF}}$ ontology $\mathcal{O}$ is a 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 finite 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 finite 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$^{\mathcal{H}}$ ontology and q an ELIQ that is $\mathcal{O}$-minimal and satisfiable w.r.t. $\mathcal{O}$. Then a frontier of q w.r.t. $\mathcal{O}$ can be computed in polynomial time.*

We note that Theorem 1 still holds when $\mathcal{O}$-minimality is dropped as a precondition and Condition 2 of frontiers is dropped as well. For proving Theorem 1, we first observe that we can concentrate on ontologies that are in normal form.

Lemma 1. *For every DL-Lite$^{\mathcal{H}}$ ontology $\mathcal{O}$, we can construct in polynomial time a DL-Lite$^{\mathcal{H}}$ ontology $\mathcal{O}'$ in normal form such that every $\mathcal{O}$-minimal ELIQ q is also $\mathcal{O}'$-minimal and a frontier of q w.r.t. $\mathcal{O}$ can be constructed in polynomial time given a frontier of q w.r.t. $\mathcal{O}'$.*

We now prove Theorem 1, adapting and generalizing a technique from [ten Cate and Dalmau, 2021]. Let $\mathcal{O}$ and $q(x_0)$ be as in the formulation of the theorem, with $\mathcal{O}$ in normal form. We may assume w.l.o.g. that q is $\mathcal{O}$-saturated. To construct a frontier of q w.r.t. $\mathcal{O}$, we consider all ways to generalize q in a least general way where 'generalizing' means to construct from q an ELIQ q' such that $q \subseteq_{\mathcal{O}} q'$ and $q' \not\subseteq_{\mathcal{O}} q$ and 'least general way' that there is no ELIQ $\widehat{q}$ that generalizes q and satisfies $\widehat{q} \subseteq_{\mathcal{O}} q'$ and $q' \not\subseteq_{\mathcal{O}} \widehat{q}$. We do this in two steps: the actual generalization plus a compensation step, the latter being needed to guarantee that we indeed arrive at a least general generalization.

For $x \in \mathsf{var}(q)$, we use q_x to denote the ELIQ obtained from q by taking the subtree of q rooted at x and making x the answer variable. The construction that follows involves the introduction of fresh variables x, some of which are a 'copy' of a variable from $\mathsf{var}(q)$. We then use $x^{\downarrow}$ to denote that original variable.

Step 1: Generalize. For each variable $x \in \mathsf{var}(q)$, define a set $\mathcal{F}_0(x)$ that contains all ELIQs which can be obtained by starting with $q_x(x)$ and then doing one of the following:

(A) *Drop concept atom*:

1. choose an atom $A(x) \in q$ such that
   (a) there is no $B(x) \in q$ with $\mathcal{O} \models B \sqsubseteq A$ and $\mathcal{O} \not\models A \sqsubseteq B$ and
   (b) there is no $R(x, y) \in q$ with $\mathcal{O} \models \exists R \sqsubseteq A$;
2. remove all $B(x) \in q$ with $\mathcal{O} \models A \equiv B$, including $A(x)$.

(B) *Generalize subquery:*

1. choose an atom $R(x, y) \in q$ directed away from x_0;
2. remove $R(x, y)$ and all atoms of q_y;
3. for each $q'(y) \in \mathcal{F}_0(y)$, add a disjoint copy $\widehat{q}'$ of q' and the role atom $R(x, y'')$ with y'' the copy of y in $\widehat{q}'$;

4. for every role S with $\mathcal{O} \models R \sqsubseteq S$ and $\mathcal{O} \not\models S \sqsubseteq R$, add a disjoint copy $\widehat{q}_y$ of q_y and the role atom $S(x, y')$ with y' the copy of y in $\widehat{q}_y$.

The definition of $x^\downarrow$ should be clear in all cases. In Point 3 of Case (B), for example, for every variable z in q' that was renamed to z' in $\widehat{q}'$ set $z'^\downarrow = z^\downarrow$. Note that $z^\downarrow$ is defined for all variables z that occur in queries in $\mathcal{F}_0(x)$. Also note that, in Point 1b of (A), it is important to use q rather than q_x as y could be a predecessor of x in q.

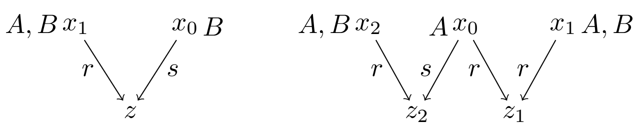
Step 2: Compensate. We construct a frontier $\mathcal{F}$ of $q(x_0)$ by including, for each $p \in \mathcal{F}_0(x_0)$, the ELIQ obtained from p by the following two steps. We write $x \rightsquigarrow^R_{q,\mathcal{O}} A$ if $\mathcal{A}_q, \mathcal{O} \models \exists R.A(x)$ and there is no $S(x, y) \in q$ with $\mathcal{O} \models S \sqsubseteq R$ and $\mathcal{A}_q, \mathcal{O} \models A(y)$.

Step 2A. Consider all $x \in \mathsf{var}(p)$, roles R, S, and concept names A such that $x^\downarrow \rightsquigarrow^R_{q,\mathcal{O}} A$, $\mathcal{O} \models R \sqsubseteq S$, and $\mathcal{O} \models \exists S \sqsubseteq B$ implies $B(x) \in p$ for all concept names B. Add the atoms $S(x, z), A(z), R(x', z)$ where z and x' are fresh variables with $z^\downarrow$ undefined, $x'^\downarrow = x^\downarrow$, and add a disjoint copy $\widehat{q}$ of q, glue the copy of $x^\downarrow$ in $\widehat{q}$ to x'.

Step 2B. Consider every $S(x, y) \in p$ directed away from x_0 that was not added in Step 2A. Then $x^\downarrow$ and $y^\downarrow$ are defined. For every role R with $\mathcal{A}_q, \mathcal{O} \models R(x^\downarrow, y^\downarrow)$, add an atom $R(z, y)$, z a fresh variable with $z^\downarrow = x^\downarrow$, as well as a disjoint copy $\widehat{q}$ of q and glue the copy of $x^\downarrow$ in $\widehat{q}$ to z.

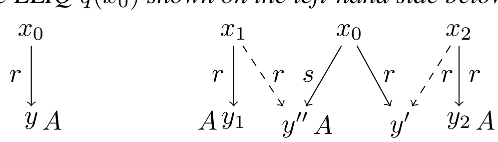
This finishes the construction of the frontier $\mathcal{F}$ of q.

Example 1. *Consider the DL-Lite$^\mathcal{H}$ ontology $\mathcal{O} = \{A \sqsubseteq \exists r,\ \exists r \sqsubseteq A,\ r \sqsubseteq s\}$ and the ELIQ $q(x_0) = A(x_0) \wedge B(x_0)$. Then $\mathcal{F}$ contains the ELIQs p_1 and p_2 shown below:*



ELIQ p_1 is the result of dropping the concept atom $A(x_0)$ and p_2 is the result of dropping the concept atom $B(x_0)$. Step 2A adds an r-successor and an s-successor of x_0 in p_2 but only an s-successor in p_1 as $\mathcal{O} \models \exists r \sqsubseteq A$, and then attaches copies of q. Step 2B does nothing, as all role atoms have been added in Step 2A.[2]

Example 2. *Consider the DL-Lite$^\mathcal{H}$ ontology $\mathcal{O} = \{r \sqsubseteq s\}$ and the ELIQ $q(x_0)$ shown on the left-hand side below:*



Then $\mathcal{F}$ contains only the ELIQ p shown on the right-hand side. It is the result of dropping the concept atom $A(y)$ in q_y, then generalizing the subquery $r(x_0, y)$ in $q_{x_0} = q$, and then compensating. Step 2A of compensation adds nothing. Step 2B adds the two dashed role atoms and attaches copies of q to x_1 and x_2.

[2] Variables x_2 and z_2 can be dropped from p_2 resulting in an ELIQ that is equivalent w.r.t. $\mathcal{O}$. We did not include such optimizations in the compensation step to avoid making it more complicated.

Lemma 2. *$\mathcal{F}$ is a frontier of $q(x_0)$ w.r.t. $\mathcal{O}$.*

We next show that the constructed frontier is of polynomial size and that its computation takes only polynomial time.

Lemma 3. *The construction of $\mathcal{F}$ runs in time polynomial in $||q|| + ||\mathcal{O}||$ (and thus $\sum_{p \in \mathcal{F}} ||p||$ is polynomial in $||q|| + ||\mathcal{O}||$).*

We next observe that adding conjunction to *DL-Lite* destroys polynomial frontiers and thus Theorem 1 does not apply to *DL-Lite*$_{\mathsf{horn}}$ ontologies [Artale *et al.*, 2009]. 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$^\wedge$)* is a unary CQ of the form $q(x_0) = A_1(x_0) \wedge \cdots \wedge A_n(x_0)$ 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$^\wedge$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.*

4 Frontiers in *DL-Lite*$^\mathcal{F}$

We start by observing that frontiers of ELIQs w.r.t. *DL-Lite*$^\mathcal{F}$ ontologies may be infinite. This leads us to identifying a syntactic restriction on *DL-Lite*$^\mathcal{F}$ ontologies that regains finite frontiers. In fact, we show that they are of polynomial size and can be computed in polynomial time.

Theorem 3. *There is an ELIQ q and a DL-Lite$^\mathcal{F}$ ontology $\mathcal{O}$ such that q does not have a finite frontier w.r.t. $\mathcal{O}$.*

In the proof of Theorem 3, we use the ELIQ $A(x)$ and

$$\mathcal{O} = \{\ A \sqsubseteq \exists r, \quad \exists r^- \sqsubseteq \exists r, \quad \exists r \sqsubseteq \exists s, \quad \mathsf{func}(r^-)\ \}.$$

The universal model $\mathcal{U}_{q,\mathcal{O}}$ of $\mathcal{A}_q$ and $\mathcal{O}$ is an infinite r-path on which every point has an s-successor. Now consider the following ELIQs $q_1, q_2, \ldots$ that satisfy $q_i \not\subseteq_\mathcal{O} q \subseteq_\mathcal{O} q_i$:

$$\begin{aligned} q_i(x_1) = {} & r(x_1, x_2), \ldots, r(x_{n-1}, x_n), s(x_n, y), s(x'_n, y), \\ & r(x'_1, x'_2), \ldots, r(x'_{n-1}, x'_n), A(x'_1). \end{aligned}$$

Any frontier $\mathcal{F}$ must contain a p_i with $p_i \subseteq_\mathcal{O} q_i$ for all $i \geq 1$. We show that, consequently, there is no bound on the size of the queries in $\mathcal{F}$. We invite the reader to apply the frontier construction from Section 3 after dropping $\mathsf{func}(r^-)$.

The proof actually shows that there is no finite frontier even if we admit the use of unrestricted CQs in the frontier in place of ELIQs. To regain finite frontiers, we restrict our attention to *DL-Lite*$^\mathcal{F}$ ontologies $\mathcal{O}$ such that if $B \sqsubseteq C$ is a CI in $\mathcal{O}$, then C contains no subconcept of the form $\exists R.D$ with $\mathsf{func}(R^-) \in \mathcal{O}$. We call such an ontology a *DL-Lite*$^{\mathcal{F}-}$ ontology. We again concentrate on ontologies in normal form.

Lemma 4. *For every DL-Lite$^{\mathcal{F}-}$ ontology $\mathcal{O}$, we can construct in polynomial time a DL-Lite$^{\mathcal{F}-}$ 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 main result of this section is as follows.

Theorem 4. *Let $\mathcal{O}$ be a DL-Lite$^{\mathcal{F}-}$ 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.*

To prove Theorem 4, let $\mathcal{O}$ and q be as in the theorem, $\mathcal{O}$ in normal form. We may assume w.l.o.g. that q is $\mathcal{O}$-minimal and $\mathcal{O}$-saturated. The construction of a frontier follows the same general approach as for *DL-Lite*$^{\mathcal{H}}$, but the presence of functional roles significantly complicates the compensation step. As before, we introduce fresh variables and rely on the mapping $x^{\downarrow}$.

Step 1: Generalize. For each variable $x \in \mathsf{var}(q)$, define a set $\mathcal{F}_0(x)$ that contains all ELIQs which can be obtained by starting with $q_x(x)$ and then doing one of the following:

(A) *Drop concept atom*: exactly as for *DL-Lite*$^{\mathcal{H}}$.

(B) *Generalize subquery:*

1. choose an atom $R(x, y) \in q$ directed away from x_0;
2. remove $R(x, y)$ and all atoms of q_y;
3. if $\mathsf{func}(R) \notin \mathcal{O}$, then for each $q'(y) \in \mathcal{F}_0(y)$ add a disjoint copy $\widehat{q}'$ of q' and the role atom $R(x, y')$ with y' the copy of y in $\widehat{q}'$;
4. if $\mathsf{func}(R) \in \mathcal{O}$ and $\mathcal{F}_0(y) \neq \emptyset$, then choose and add a $q' \in \mathcal{F}_0(y)$ and the role atom $R(x, y)$.

Step 2: Compensate. We construct a frontier $\mathcal{F}$ of $q(x_0)$ by including, for each $p \in \mathcal{F}_0(x_0)$, the CQ obtained from p by the following two steps. For $x \in \mathsf{var}(q)$, R a role, and M a set of concept names from $\mathcal{O}$, we write $x \rightsquigarrow^{R}_{q,\mathcal{O}} M$ if M is maximal with $\mathcal{A}_q, \mathcal{O} \models \exists R. \bigsqcap M(x)$ and there is no $R(x, y) \in q$ with $\mathcal{A}_q, \mathcal{O} \models \bigsqcap M(y)$.

Step 2A. Consider every $x \in \mathsf{var}(p)$, role R, and set of concept names $M = \{A_1, \ldots, A_k\}$ with $x^{\downarrow} \rightsquigarrow^{R}_{q,\mathcal{O}} M$. If $\mathcal{O} \models \exists R \sqsubseteq B$ implies $B(x) \in p$ for all concept names B, add the atoms $R(x, z), A_1(z), \ldots, A_k(z)$ where z is a fresh variable, and leave $z^{\downarrow}$ undefined.

Step 2B. This step is iterative. For bookkeeping, we mark atoms $R(x, y) \in p$ to be processed in the next round of the iteration. Marking is only applied to atoms $R(x, y)$ directed away from x_0 such that $y^{\downarrow}$ is defined and if $x^{\downarrow}$ is undefined then $\mathsf{func}(R^-) \notin \mathcal{O}$ or q contains no atom of the form $R(y^{\downarrow}, z)$.

Start. Consider every $R(x, y) \in p$ directed away from x_0 with $\mathsf{func}(R^-) \notin \mathcal{O}$. Then $x^{\downarrow}$ is defined. Extend p with atom $R^-(y, x')$, x' a fresh variable with $x'^{\downarrow} = x^{\downarrow}$. Mark the new atom.

Step. Choose a marked atom $R(x, y)$ and unmark it. If $\mathsf{func}(R^-) \notin \mathcal{O}$ or q contains no atom of the form $R(y^{\downarrow}, z)$, then add a disjoint copy $\widehat{q}$ of q and glue the copy of $y^{\downarrow}$ in $\widehat{q}$ to y. Otherwise, do the following:

(i) add $A(y)$ whenever $\mathcal{A}_q, \mathcal{O} \models A(y^{\downarrow})$;

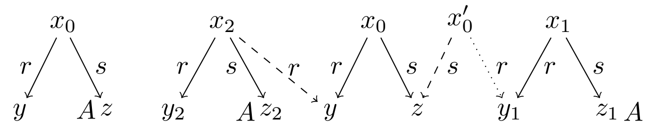
(ii) for all atoms $S(y^{\downarrow}, z) \in q$ with $S(y^{\downarrow}, z) \neq R^-(y^{\downarrow}, x^{\downarrow})$, extend p with atom $S(y, z')$, z' a fresh variable with $z'^{\downarrow} = z$. Mark $S(y, z')$.

(iii) For all roles S and sets $M = \{A_1, \ldots, A_k\}$ such that $y^{\downarrow} \rightsquigarrow^{S}_{q,\mathcal{O}} M$, extend p with atoms $S(y, u)$, $S^-(u, y'), A_1(u), \ldots, A_k(u)$ where u and y' are fresh variables. Set $y'^{\downarrow} = y^{\downarrow}$ and mark $S^-(u, y')$.

The step is repeated as long as possible. Note that in Point (iii), the role S must occur on the right-hand side of some CI in the *DL-Lite*$^{\mathcal{F}-}$ ontology $\mathcal{O}$. Consequently, $\mathsf{func}(S^-) \notin \mathcal{O}$ and it is not a problem that u receives two S-predecessors. Also in Point (iii), $\mathsf{func}(S) \in \mathcal{O}$ implies that q cannot contain an atom $S(y^{\downarrow}, z)$ due to the definition of '$\rightsquigarrow$' and thus we may leave $u^{\downarrow}$ undefined.

This finishes the construction of the frontier $\mathcal{F}$ of q.

Example 3. *Consider the ontology $\mathcal{O} = \{\mathsf{func}(s)\}$ and ELIQ $q(x_0)$ shown on the left-hand side below:*



The ELIQ $p \in \mathcal{F}$ shown on the right-hand side is the result of dropping the concept atom $A(z)$ in q_z, then generalizing the subquery $s(x_0, z)$ in $q_{x_0} = q$, and then compensating. Step 2A of compensation adds nothing. The start of Step 2B adds the two dashed role atoms and marks them. The step of Step 2B adds the dotted role atom via Point (ii) and marks it. When the step of Step 2B processes role atoms $r^-(y, x_2)$ and $r(x_0', y)$, it attaches copies of q to x_2 and y_1. Note that directly attaching a copy of q to x_0' would violate $\mathsf{func}(s)$.

Lemma 5. *$\mathcal{F}$ is a frontier of $q(x_0)$ w.r.t. $\mathcal{O}$.*

As for *DL-Lite*$^{\mathcal{H}}$, the constructed frontier is of polynomial size and its computation takes only polynomial time. Crucially, the iterative process in Point 2B terminates since in Step (ii) a (copy of a) subquery of q is added and the process stops at atoms added in Step (iii).

Lemma 6. *The construction of $\mathcal{F}$ runs in time polynomial in $||q|| + ||\mathcal{O}||$ (and thus $\sum_{p \in \mathcal{F}} ||p||$ is polynomial in $||q|| + ||\mathcal{O}||$).*

5 Uniquely Characterizing ELIQs

As a first application of our results on frontiers, we consider the unique characterization of ELIQs in terms of polynomially many data examples. One area where this is relevant is reverse query engineering, also known as *query-by-example (QBE)*, which in a DL context was studied in [Gutiérrez-Basulto *et al.*, 2018; Funk *et al.*, 2019; Ortiz, 2019]. The idea of QBE is that a query is not formulated directly, but derived from data examples that describe its behavior. The results in this section imply that every ELIQ can be described up to equivalence by such examples this is always possible and that a reasonable number of examples of reasonable size suffices.

Formally, 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. We say that an ELIQ q *fits* (E^+, E^-) *w.r.t. a DL-Lite*$^{\mathcal{HF}}$*ontology* $\mathcal{O}$ if $(\mathcal{A}, a) \in E^+$ implies $\mathcal{A}, \mathcal{O} \models q(a)$

Algorithm 1 Learning ELIQs under *DL-Lite* ontologies

Input An ontology $\mathcal{O}$ in normal form 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

and $(\mathcal{A}, a) \in E^-$ implies $\mathcal{A}, \mathcal{O} \not\models q(a)$. Then (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 Theorems 1 and 4, see also [ten Cate and Dalmau, 2021].

Theorem 5. *Let $\mathcal{O}$ be an ontology formulated in DL-Lite$^\mathcal{H}$ or DL-Lite$^{\mathcal{F}-}$. Then for every ELIQ q that is satisfiable w.r.t. $\mathcal{O}$, there are sets of data examples (E^+, E^-) that uniquely characterize q w.r.t. $\mathcal{O}$ and such that $||(E^+, E^-)||$ is polynomial in $||q|| + ||\mathcal{O}||$. If $\mathcal{O}$ is a DL-Lite$^{\mathcal{F}-}$ ontology, then (E^+, E^-) can be computed in polynomial time and the same holds for DL-Lite$^\mathcal{H}$ if q is $\mathcal{O}$-minimal.*

6 Learning ELIQs under Ontologies

We use our results on frontiers to show that ELIQs are polynomial time learnable under ontologies formulated in *DL-Lite*$^\mathcal{H}$ and *DL-Lite*$^{\mathcal{F}-}$, using only membership queries. We also present two results on non-learnability.

Theorem 6. *ELIQs are polynomial time learnable under DL-Lite$^\mathcal{H}$ ontologies and under DL-Lite$^{\mathcal{F}-}$ ontologies using only membership queries.*

If the ontology contains concept disjointness constraints, then this only holds true if the learner is provided with a seed CQ (definition given below).

For proving Theorem 6, let $\mathcal{O}$ be an ontology formulated in *DL-Lite*$^\mathcal{H}$ or *DL-Lite*$^{\mathcal{F}-}$ and $q_T(x_0)$ the target ELIQ known to the oracle. We may again assume $\mathcal{O}$ to be in normal form.

Lemma 7. *In DL-Lite$^\mathcal{H}$ and DL-Lite$^{\mathcal{F}-}$, every polynomial time learning algorithm for ELIQs under ontologies in normal form that uses only membership queries can be transformed into a learning algorithm with the same properties for ELIQs under unrestricted ontologies.*

The learning algorithm is displayed as Algorithm 1. It assumes a *seed CQ* q_H^0, that is, a CQ q_H^0 such that $q_H^0 \subseteq_\mathcal{O} q_T$ and q_H^0 is satisfiable w.r.t. $\mathcal{O}$. If $\mathcal{O}$ contains no disjointness constraints, then for $\Sigma = \mathsf{sig}(\mathcal{O})$ we can use as the seed CQ

$$q_H^0(x_0) = \{A(x_0) \mid A \in \Sigma \cap \mathsf{N_C}\} \cup \{r(x_0, x_0) \mid r \in \Sigma \cap \mathsf{N_R}\}.$$

We can still construct a seed CQ q_H^0 in time polynomial in $||\mathcal{O}||$ if $\mathcal{O}$ contains no disjoint constraints on concepts (but potentially on roles); details are in the appendix. In the presence of concept disjointness constraints, a seed CQ can be obtained through an initial equivalence query.

The algorithm constructs and repeatedly updates a hypothesis ELIQ q_H while maintaining the invariant $q_H \subseteq_\mathcal{O} q_T$. The initial call to subroutine 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 $q_H \subseteq_\mathcal{O} q_T$. In between, the algorithm applies the minimize subroutine to ensure that the new q_H is $\mathcal{O}$-minimal and to avoid an excessive blowup while iterating in the while loop.

We next detail the subroutines treeify and minimize. We define minimize on unrestricted CQs since it is applied to non-ELIQs as part of the treeify subroutine.

The minimize **subroutine.** The subroutine takes as input a unary CQ $q(x_0)$ that is satisfiable w.r.t. $\mathcal{O}$ and satisfies $q \subseteq_\mathcal{O} q_T$. It computes a unary CQ q' with $q \subseteq_\mathcal{O} q' \subseteq_\mathcal{O} q_T$ using membership queries that is minimal in a strong sense. Formally, minimize first makes sure that q is $\mathcal{O}$-saturated and then exhaustively applies the following operation:

Remove atom. Choose a role atom $r(x, y) \in q$ and let q^- be the maximal connected component of $q \setminus \{r(x, y)\}$ that contains x_0. Pose the membership query $\mathcal{A}_{q^-}, \mathcal{O} \models q_T(x_0)$. If the response is positive, continue with q^- in place of q.

Clearly, the result of minimize is $\mathcal{O}$-minimal.

The treeify **subroutine.** The subroutine takes as input a unary CQ $q(x_0)$ that is satisfiable w.r.t. $\mathcal{O}$, and satisfies $q \subseteq_\mathcal{O} q_T$. It computes an ELIQ q' with $q \subseteq_\mathcal{O} q' \subseteq_\mathcal{O} q_T$ by repeatedly increasing the length of cycles in q and minimizing the obtained query; a similar construction is used in [ten Cate and Dalmau, 2021]. The resulting ELIQ is $\mathcal{O}$-minimal.

Formally, treeify first makes sure that $q(x_0)$ is $\mathcal{O}$-saturated and then constructs a sequence of CQs $p_1, p_2, \dots$ 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 some cycle. Here, a *cycle* in a CQ q is a sequence $R_1(x_1, x_2), \dots, R_n(x_n, x_1)$ of distinct role atoms in q such that $x_1, \dots x_n$ are distinct. More precisely, p_i' is the result of the following operation.

Double cycle. Choose a role atom $r(x, y) \in p_i$ that is part of a cycle in p_i and let p be $p_i \setminus \{r(x, y)\}$. The CQ p_i' is then obtained by starting with p, adding a disjoint copy p' of p where x' refers to the copy of $x \in \mathsf{var}(p)$ in p' and adding the role atoms $r(x, y'), r(x', y)$.

If p_i contains no more cycles, treeify stops and returns p_i.

Returning to Algorithm 1, let $q_1, q_2, \dots$ be the sequence of ELIQs that are assigned to q_H during a run of the learning algorithm. We show in the appendix that for all $i \geq 1$, it holds that $q_i \subseteq_\mathcal{O} q_T$, $q_i \subseteq_\mathcal{O} q_{i+1}$ while $q_{i+1} \not\subseteq_\mathcal{O} q_i$, and $|\mathsf{var}(q_{i+1})| \geq |\mathsf{var}(q_i)|$. This can be used to prove that the while loop in Algorithm 1 terminates after a polynomial number of iterations, arriving at a hypothesis q_H with $q_H \equiv_\mathcal{O} q_T$.

We now turn to non-learnability results. Without a seed CQ, ontologies with concept disjointness constraints are not learnable using only polynomially many membership queries. A *disjointness ontology* is a *DL-Lite*$^{\mathcal{HF}}$ ontology that only consists of concept disjointness constraints.

Theorem 7. *$AQ^\wedge$s are not learnable under disjointness ontologies using only polynomially many membership queries.*

We next show that when we drop the syntactic restriction from $DL\text{-}Lite^{\mathcal{F}-}$, then ELIQs are no longer learnable at all using only membership queries. Note that this is not a direct consequence of Theorem 3 as there could be an alternative approach that does not use frontiers.

Theorem 8. *ELIQs are not learnable under $DL\text{-}Lite^{\mathcal{F}}$ ontologies using only membership queries.*

7 Outlook

A natural next step for future work is to generalize the results presented in this paper to $DL\text{-}Lite^{\mathcal{HF}}$, adopting the same syntactic restriction that we have adopted for $DL\text{-}Lite^{\mathcal{F}}$, and additionally requiring that functional roles have no proper subroles. The latter serves to control the interaction between functional roles and role inclusions. Even with this restriction, however, that interaction is very subtle and the frontier construction becomes significantly more complex. Other interesting questions are whether ELIQs can be learned in polynomial time w.r.t. $DL\text{-}Lite_{\mathsf{horn}}$ ontologies and whether CQs can be learned w.r.t. $DL\text{-}Lite_{\mathsf{core}}$ ontologies when both membership and equivalence queries are admitted.

Acknowledgements

Carsten Lutz was supported by DFG CRC 1320 EASE.

References

[Alspach *et al.*, 1987] Brian Alspach, Jean-Claude Bermond, and Dominique Sotteau. Decomposition into cycles I: Hamilton decompositions. In *Proc. of the Workshop on Cycles and Rays*, pages 9–18, 1987.

[Angluin, 1987a] Dana Angluin. Learning regular sets from queries and counterexamples. *Inf. Comput.*, 75(2):87–106, 1987.

[Angluin, 1987b] Dana Angluin. Queries and concept learning. *Mach. Learn.*, 2(4):319–342, 1987.

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

[Baader *et al.*, 1999] Franz Baader, Ralf Küsters, and Ralf Molitor. Computing least common subsumers in description logics with existential restrictions. In *Proc. of IJCAI*, pages 96–103. Morgan Kaufmann, 1999.

[Baader *et al.*, 2007] Franz Baader, Baris Sertkaya, and Anni-Yasmin Turhan. Computing the least common subsumer w.r.t. a background terminology. *J. Appl. Log.*, 5(3):392–420, 2007.

[Baader *et al.*, 2017] Franz Baader, Ian Horrocks, Carsten Lutz, and Ulrike Sattler. *An Introduction to Description Logics*. Cambride University Press, 2017.

[Baader *et al.*, 2018] Franz Baader, Francesco Kriegel, Adrian Nuradiansyah, and Rafael Peñaloza. Making repairs in description logics more gentle. In *Proc. of KR*, pages 319–328. AAAI Press, 2018.

[Baader, 2003] Franz Baader. Least common subsumers and most specific concepts in a description logic with existential restrictions and terminological cycles. In *Proc. of IJCAI*, pages 319–324. Morgan Kaufmann, 2003.

[Bienvenu *et al.*, 2013a] Meghyn Bienvenu, Magdalena Ortiz, Mantas Simkus, and Guohui Xiao. Tractable queries for lightweight description logics. In *Proc. of IJCAI*, pages 768–774, 2013.

[Bienvenu *et al.*, 2013b] Meghyn Bienvenu, Balder ten Cate, Carsten Lutz, and Frank Wolter. Ontology-based data access: a study through disjunctive datalog, CSP, and MMSNP. In *Proc. of PODS*, pages 213–224. ACM, 2013.

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

[Calvanese *et al.*, 2009] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Mariano Rodriguez-Muro, and Riccardo Rosati. Ontologies and databases: The DL-Lite approach. In *Reasoning Web*, volume 5689 of *LNCS*, pages 255–356, 2009.

[Funk *et al.*, 2019] Maurice Funk, Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. Learning description logic concepts: When can positive and negative examples be separated? In *Proc. of IJCAI*, pages 1682–1688, 2019.

[Funk *et al.*, 2021] Maurice Funk, Jean Christoph Jung, and Carsten Lutz. Actively learning concept and conjunctive queries under $\mathcal{EL}^r$-ontologies. In *Proc. of IJCAI*, pages 1887–1893, 2021.

[Gutiérrez-Basulto *et al.*, 2018] Víctor Gutiérrez-Basulto, Jean Christoph Jung, and Leif Sabellek. Reverse engineering queries in ontology-enriched systems: The case of expressive Horn description logic ontologies. In *Proc. of IJCAI-ECAI*, pages 1847–1853. ijcai.org, 2018.

[Jung *et al.*, 2020a] Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. Logical separability of incomplete data under ontologies. In *Proc. of KR*, pages 517–528, 2020.

[Jung *et al.*, 2020b] Jean Christoph Jung, Carsten Lutz, and Frank Wolter. Least general generalizations in description logic: Verification and existence. In *Proc. of AAAI*, pages 2854–2861, 2020.

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

[Konev *et al.*, 2018] Boris Konev, Carsten Lutz, Ana Ozaki, and Frank Wolter. Exact learning of lightweight description logic ontologies. *J. Mach. Learn. Res.*, 18(201):1–63, 2018.

[Kriegel, 2019] Francesco Kriegel. *Constructing and Extending Description Logic Ontologies using Methods of Formal Concept Analysis*. PhD thesis, TU Dresden, 2019.

[Lehmann and Hitzler, 2010] Jens Lehmann and Pascal Hitzler. Concept learning in description logics using refinement operators. *Mach. Learn.*, 78:203–250, 2010.

[Lehmann and Völker, 2014] Jens Lehmann and Johanna Völker. *Perspectives on Ontology Learning*, volume 18 of *Studies on the Semantic Web*. IOS Press, 2014.

[Muggleton, 1991] Stephen Muggleton. Inductive logic programming. *New Generation Comput.*, 8(4):295–318, 1991.

[Nesetril and Tardif, 2000] Jaroslav Nesetril and Claude Tardif. Duality theorems for finite structures (characterising gaps and good characterisations). *J. Comb. Theory, Ser. B*, 80(1):80–97, 2000.

[Ortiz, 2019] Magdalena Ortiz. Ontology-mediated queries from examples: a glimpse at the DL-Lite case. In *Proc. of GCAI*, pages 1–14, 2019.

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

[Ozaki *et al.*, 2020] Ana Ozaki, Cosimo Persia, and Andrea Mazzullo. Learning query inseparable $\mathcal{ELH}$ ontologies. In *Proc. of AAAI*, pages 2959–2966, 2020.

[Ozaki, 2020] Ana Ozaki. Learning description logic ontologies: Five approaches. where do they stand? *KI - Künstliche Intelligenz*, 2020.

[Plotkin, 1970] Gordon Plotkin. A note on inductive generalizations. *Edinburgh University Press*, 1970.

[Sarker and Hitzler, 2019] Md. Kamruzzaman Sarker and Pascal Hitzler. Efficient concept induction for description logics. In *Proc. of AAAI*, pages 3036–3043, 2019.

[ten Cate and Dalmau, 2021] Balder ten Cate and Victor Dalmau. Conjunctive queries: Unique characterizations and exact learnability. In *Proc. of ICDT*, volume 186 of *LIPIcs*, pages 9:1–9:24. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[Zarrieß and Turhan, 2013] Benjamin Zarrieß and Anni-Yasmin Turhan. Most specific generalizations w.r.t. general $\mathcal{EL}$-TBoxes. In *Proc. of IJCAI*, pages 1191–1197, 2013.

A Additional Preliminaries

We introduce some additional preliminaries that are needed for the lemmas and proofs in the appendix.

We start with defining the semantics of conjunctive queries in full detail. 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 CQ $q(x_0)$, we write $q(x_0) \to (\mathcal{I}, d)$ if there is a homomorphism h from q to $\mathcal{I}$ with $h(x_0) = d$. Let $q(x_0)$ be a 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 $q(x_0) \to (\mathcal{I}, d)$. Now let $\mathcal{O}$ be an 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 $\mathcal{A}$.

Let $q(x_0)$ be an ELIQ. The *codepth* of a variable $x \in \mathsf{var}(q)$ is 0 if there is no $R(x, x') \in q$ directed away from x_0 and is $k+1$ if k is the maximum of the codepths of all $x' \in \mathsf{var}(q)$ with $R(x,x') \in q$ directed away from x_0.

We next define universal models, first for *DL-Lite*$^{\mathcal{H}}$ and then for *DL-Lite*$^{\mathcal{F}}$. Let $\mathcal{O}$ be a *DL-Lite*$^{\mathcal{H}}$ ontology and let $\mathcal{A}$ be an ABox that is satisfiable w.r.t. $\mathcal{O}$. For $a \in \mathsf{ind}(\mathcal{A})$, concept names A, and R a role, we write $a \rightsquigarrow^{R}_{\mathcal{A},\mathcal{O}} A$ if $\mathcal{A}, \mathcal{O} \models \exists R.A(a)$ and there is no $S(a,b) \in \mathcal{A}$ such that $\mathcal{O} \models S \sqsubseteq R$ and $\mathcal{A}, \mathcal{O} \models A(b)$. Note that this is identical to the definition of '$\rightsquigarrow$' given in Section 3, but is formulated for ABoxes in place of ELIQs.

A *trace* for $\mathcal{A}$ and $\mathcal{O}$ is a sequence $t = aR_1A_1R_2A_2\dots R_nA_n$, $n \geq 0$ where $a \in \mathsf{ind}(\mathcal{A})$, $R_1, \dots, R_n$ are roles that occur in $\mathcal{O}$, and $A_1, \dots, A_n$ are sets of concept names that occur in $\mathcal{O}$, such that $a \rightsquigarrow^{R_1}_{\mathcal{A},\mathcal{O}} A_1$ and $\mathcal{O} \models A_i \sqsubseteq \exists R_{i+1}.A_{i+1}$ 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

$$\begin{aligned}\mathcal{U}_{\mathcal{A},\mathcal{O}} = \mathcal{A} \cup \{&A(a) \mid \mathcal{A}, \mathcal{O} \models A(a)\} \cup \\ &\{S(a,b) \mid R(a,b) \in \mathcal{A} \text{ and } \mathcal{O} \models R \sqsubseteq S\} \cup \\ &\{B(tRA) \mid tRA \in \mathbf{T} \text{ and } \mathcal{O} \models A \sqsubseteq B\} \cup \\ &\{S(t, tRA) \mid tRA \in \mathbf{T} \text{ and } \mathcal{O} \models R \sqsubseteq S\}.\end{aligned}$$

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

Now let $\mathcal{O}$ be a *DL-Lite*$^{\mathcal{F}}$ ontology and let $\mathcal{A}$ be an ABox that is satisfiable w.r.t. $\mathcal{O}$. For a set M of concept names, we write $\bigsqcap M$ as a shorthand for $\bigsqcap_{A \in M} A$. For $a \in \mathsf{ind}(\mathcal{A})$, M, M' sets of concept names, and R a role, we write

- $a \rightsquigarrow^{R}_{\mathcal{A},\mathcal{O}} M$ if $\mathcal{A}, \mathcal{O} \models \exists R.\bigsqcap M(a)$, M is maximal with this condition, and there is no $R(a,b) \in \mathcal{A}$ such that $\mathcal{A}, \mathcal{O} \models \bigsqcap M(b)$;
- $M \rightsquigarrow^{R}_{\mathcal{O}} M'$ if $\mathcal{O} \models \bigsqcap M \sqsubseteq \exists R.\bigsqcap M'(a)$ and M' is maximal with this condition.

The definition of '$\rightsquigarrow$' in the first item is identical to the definition of '$\rightsquigarrow$' given in Section 4, but is formulated for ABoxes in place of ELIQs. The maximality of M is important when dealing with functional roles. If, for example, $\mathcal{O}$ contains $A \sqsubseteq \exists r.B_1$, $A \sqsubseteq \exists r.B_2$, and $\mathsf{func}(r)$ and $\mathcal{A} = \{A(a)\}$, then we have $a \rightsquigarrow^r_{\mathcal{A},\mathcal{O}} \{B_1, B_2\}$, but not $a \rightsquigarrow^r_{\mathcal{A},\mathcal{O}} \{B_i\}$ for an $i \in \{1, 2\}$. This helps to ensure that a gets only a single r-successor in the universal model.

A *trace* for $\mathcal{A}$ and $\mathcal{O}$ is a sequence $t = aR_1M_1R_2M_2 \dots R_nM_n$, $n \geq 0$ where $a \in \mathsf{ind}(\mathcal{A})$, $R_1, \dots, R_n$ are roles that occur in $\mathcal{O}$, and $M_1, \dots, M_n$ are sets of concept names that occur in $\mathcal{O}$, such that

(i) $a \rightsquigarrow^{R_1}_{\mathcal{A},\mathcal{O}} M_1$ and

(ii) for $1 \leq i < n$, we have $M_i \rightsquigarrow^{R_{i+1}}_{\mathcal{O}} M_{i+1}$ and if $\mathsf{func}(R_i^-) \in \mathcal{O}$, then $R_{i+1} \neq R_i^-$.

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 defined as

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

In the following, we give three elementary lemmas that pertain to universal models and are used throughout the appendix. Their proof is entirely standard and omitted. The most important properties of universal models are as follows.

Lemma 8. *Let $\mathcal{O}$ be an ontology formulated in DL-Lite$^{\mathcal{H}}$ or DL-Lite$^{\mathcal{F}}$ and let $\mathcal{A}$ 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 $\bar{a} \in \mathsf{ind}(\mathcal{A})^{|\bar{x}|}$.*

We note that Lemma 8 ceases to hold when $\mathcal{O}$ is formulated in *DL-Lite*$^{\mathcal{HF}}$ due to subtle interactions between role inclusions and functionality assertions.

The next lemma links query containment to the existence of homomorphisms into the universal model.

Lemma 9. *Let $\mathcal{O}$ be an ontology formulated in DL-Lite$^{\mathcal{H}}$ or DL-Lite$^{\mathcal{F}}$, and let $q_1(\bar{x})$, $q_2(\bar{y})$ be CQs that are satisfiable w.r.t. $\mathcal{O}$. Then $q_1 \subseteq_{\mathcal{O}} q_2$ iff $q_2(\bar{y}) \to (\mathcal{U}_{q_1,\mathcal{O}}, \bar{x})$.*

The following lemma states that any homomorphism from a CQ q to some universal model $\mathcal{U}_{\mathcal{A},\mathcal{O}}$ can be extended to a homomorphism from $\mathcal{U}_{q,\mathcal{O}}$ to $\mathcal{U}_{\mathcal{A},\mathcal{O}}$.

Lemma 10. *Let $\mathcal{O}$ be an ontology formulated in DL-Lite$^{\mathcal{H}}$ or DL-Lite$^{\mathcal{F}}$, $\mathcal{A}$ an ABox, and $q(x_0)$ 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 $h(x_0) = a$ for some $a \in \mathsf{ind}(\mathcal{A})$, then h can be extended to a homomorphism h' from $\mathcal{U}_{q,\mathcal{O}}$ to $\mathcal{U}_{\mathcal{A},\mathcal{O}}$ with $h'(x_0) = a$.*

And finally, we show a property of $\mathcal{O}$-minimal and $\mathcal{O}$-saturated queries that we will use to show that the constructions indeed yield frontiers. Note that the property implies that every homomorphism h from an ELIQ $q(x_0)$ to $\mathcal{U}_{q,\mathcal{O}}$ with $h(x_0) = x_0$ is injective.

Lemma 11. *Let $\mathcal{O}$ be an ontology in normal form formulated in DL-Lite$^{\mathcal{H}}$ or DL-Lite$^{\mathcal{F}}$, and $q(x_0)$ 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 $h(x_0) = x_0$.*

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 for all $y \in \mathsf{var}(q)$, $h(y) \notin \mathsf{var}(q_x)$, since q is connected and there is no $y' \in \mathsf{var}(q)$ with $h(y') = x$.

Next let $yR_1M_1 \dots R_nM_n \in \Delta^{\mathcal{U}_{q,\mathcal{O}}}$ be a trace starting with some variable $y \in \mathsf{var}(q')$. Then $y \rightsquigarrow^{R_1}_{q,\mathcal{O}} M_1$ and therefore $\mathcal{A}_q, \mathcal{O} \models \exists R_1.\bigsqcap M_1(y)$ and there is no $R_1(y, y') \in q$ such that $\mathcal{A}_q, \mathcal{O} \models \bigsqcap M_1(y')$. Since $\mathcal{O}$ is in normal form, there is a set of concept names M such that $\mathcal{A}_q, \mathcal{O} \models \bigsqcap M(y)$ and $\mathcal{O} \models \bigsqcap M \sqsubseteq \exists R_1.\bigsqcap M_1(y)$. By $\mathcal{O}$-saturation of q, $\mathcal{A}_{q'}, \mathcal{O} \models \bigsqcap M(y)$ and therefore $\mathcal{A}_{q'}, \mathcal{O} \models \exists R_1.\bigsqcap M_1(y)$. Since q' is a subset of q, $y \rightsquigarrow^{R_1}_{q',\mathcal{O}} M_1$ and thus $yR_1M_1 \dots R_nM_n \in \Delta^{\mathcal{U}_{q',\mathcal{O}}}$.

Let $A(y) \in q$. If $h(y) \in \mathsf{var}(q')$, then $A(h(y)) \in \mathcal{U}_{q',\mathcal{O}}$ by $\mathcal{O}$-saturation of q. If $h(y)$ is a trace, then, by connectedness it is a trace below a variable of q' and thus $A(h(y)) \in \mathcal{U}_{q',\mathcal{O}}$.

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

Thus, h is a homomorphism from q to $\mathcal{U}_{q',\mathcal{O}}$ as required. ❑

B Proofs for Section 2

We describe how to convert a *DL-Lite*$^{\mathcal{HF}}$-ontology $\mathcal{O}$ into a *DL-Lite*$^{\mathcal{HF}}$-ontology $\mathcal{O}'$ in normal form. We use $\mathfrak{C}(\mathcal{O})$ to denote the set of all concepts that occur on the right-hand side of a concept inclusion in $\mathcal{O}$. Note that $\mathfrak{C}(\mathcal{O})$ is closed under sub-concepts. We introduce a fresh concept name X_C for every complex concept $C \in \mathfrak{C}(\mathcal{O})$, and set $X_A = A$ for concept names $A \in \mathfrak{C}(\mathcal{O})$. The ontology $\mathcal{O}'$ consists of the following concept and role inclusions:

- all role inclusions from $\mathcal{O}$;
- $C \sqsubseteq X_D$ for every $C \sqsubseteq D \in \mathcal{O}$;
- $X_{D_1 \sqcap D_2} \sqsubseteq X_{D_i}$, for every $D_1 \sqcap D_2 \in \mathfrak{C}(\mathcal{O})$ and $i \in \{1, 2\}$;
- $X_{\exists R.C} \sqsubseteq \exists R.X_C$, for every $\exists R.C \in \mathfrak{C}(\mathcal{O})$;

Clearly, $\mathcal{O}'$ can be computed in polynomial time. Moreover, it is easy to verify that for all $C \in \mathfrak{C}(\mathcal{O})$, we have $\mathcal{O}' \models X_C \sqsubseteq C$. Regarding the relationship between $\mathcal{O}$ and $\mathcal{O}'$, we observe the following consequences of the definition of $\mathcal{O}'$.

Lemma 12.

1. *$\mathcal{O}' \models \mathcal{O}$, that is, every model of $\mathcal{O}'$ is a model of $\mathcal{O}$;*

2. *every model $\mathcal{I}$ of $\mathcal{O}$ can be extended into a model $\mathcal{I}'$ of $\mathcal{O}'$ by starting with $\mathcal{I}' = \mathcal{I}$ and then setting for every complex concept $C \in \mathfrak{C}(\mathcal{O})$, $X_C^{\mathcal{I}'} = C^{\mathcal{I}}$.*

Lemma 12 essentially says that $\mathcal{O}'$ is a conservative extension of $\mathcal{O}$, but is slightly stronger in also making precise how exactly a model of $\mathcal{O}$ can be extended to a model of $\mathcal{O}'$.

C Proofs for Section 3

Lemma 1. *For every DL-Lite$^{\mathcal{H}}$ ontology $\mathcal{O}$, we can construct in polynomial time a DL-Lite$^{\mathcal{H}}$ ontology $\mathcal{O}'$ in normal form such that every $\mathcal{O}$-minimal ELIQ q is also $\mathcal{O}'$-minimal and 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*$^{\mathcal{H}}$ ontology and let $\mathcal{O}'$ be the result of converting $\mathcal{O}$ into normal form as described before Lemma 12. Moreover, let $q(x)$ be an ELIQ and $\mathcal{F}'$ a frontier of q w.r.t. $\mathcal{O}'$. Let $\mathcal{F}$ be obtained from $\mathcal{F}'$ by including all ELIQs that can be obtained by taking an ELIQ $p \in \mathcal{F}'$ and then doing the following:

- for every atom $X_C(x)$, $C \in \mathfrak{C}(\mathcal{O})$, remove that atom and add a variable disjoint copy of C viewed as an ELIQ, gluing the root to x;

To prove that $\mathcal{F}$ is a frontier of q w.r.t. $\mathcal{O}$, we show that the three conditions from the definition of frontiers are satisfied:

1. $q \subseteq_{\mathcal{O}} q_F$ for all $q_F \in \mathcal{F}$.

   Assume to the contrary that there is a $q_F(x) \in \mathcal{F}$ with $q \not\subseteq_{\mathcal{O}} q_F$. 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 $\mathcal{A}, \mathcal{O} \not\models q_F(a)$. We may assume that concept names X_C do not occur in $\mathcal{A}$ as they are not used in $\mathcal{O}$, q, and q_F. From $\mathcal{O}' \models \mathcal{O}$, we obtain $\mathcal{A}, \mathcal{O}' \models q(a)$. Since $\mathcal{A}, \mathcal{O} \not\models q_F(a)$, there is a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{O}$ such that $\mathcal{I} \not\models q_F(a)$. Let $\mathcal{I}'$ be the extension of $\mathcal{I}$ according to Point 2 of Lemma 12. Then $\mathcal{I}'$ is a model of $\mathcal{O}'$ and $\mathcal{A}$. Now, let $q_F^0(x) \in \mathcal{F}'$ be the query from which $q_F(x)$ was obtained in the construction of $\mathcal{F}$. By construction of q_F and of $\mathcal{I}'$, $\mathcal{I}' \not\models q_F^0(a)$, so $\mathcal{A}, \mathcal{O}' \not\models q_F^0(a)$. Thus, $\mathcal{A}$ witnesses that $q \not\subseteq_{\mathcal{O}'} q_F^0$, a contradiction to q_F^0 being in $\mathcal{F}'$.

2. $q_F \not\subseteq q$ for all $q_F \in \mathcal{F}$.

   Let $q_F(x) \in \mathcal{F}$ and let $q_F^0(x) \in \mathcal{F}'$ be the ELIQ from which $q_F(x)$ was obtained during the construction of $\mathcal{F}$. Since $q_F^0 \not\subseteq_{\mathcal{O}'} q$, there is an ABox $\mathcal{A}'$ and an individual $a \in \mathsf{ind}(\mathcal{A}')$ such that $\mathcal{A}', \mathcal{O}' \models q_0^F(a)$, but $\mathcal{A}', \mathcal{O}' \not\models q(a)$. Let the ABox $\mathcal{A}$ be obtained by starting with $\mathcal{A}'$ and adding $C(b)$, for each concept assertion $X_C(b) \in \mathcal{A}'$. Here, the addition of $C(b)$, with C an $\mathcal{ELI}$-concept, is defined as expected: view $C(b)$ as a tree-shaped ABox $\mathcal{A}_{C(b)}$ that uses only fresh individual names, and then add this ABox gluing its root to b.

   We aim to show $\mathcal{A}, \mathcal{O} \models q_F(a)$ and $\mathcal{A}, \mathcal{O} \not\models q(a)$, witnessing $q_F \not\subseteq_{\mathcal{O}} q$ as required.

   For the former, assume to the contrary that $\mathcal{A}, \mathcal{O} \not\models q_F(a)$. Then there is a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{O}$ with $\mathcal{I} \not\models q_F(a)$. Let $\mathcal{I}'$ be the extension of $\mathcal{I}$ according to Point 2 of Lemma 12. Then $\mathcal{I}'$ is a model of $\mathcal{O}'$ and, by construction of $\mathcal{I}'$ and of $\mathcal{A}$ from $\mathcal{A}'$, also a model of $\mathcal{A}'$. Moreover, $\mathcal{I} \not\models q_F(a)$ implies $\mathcal{I}' \not\models q_F^0(a)$ by construction of q_F and of $\mathcal{I}'$. This contradicts $\mathcal{A}', \mathcal{O}' \models q_0^F(a)$.

   It remains to show that $\mathcal{A}, \mathcal{O} \not\models q(a)$. Since $\mathcal{A}', \mathcal{O}' \not\models q(a)$, there is a model $\mathcal{I}$ of $\mathcal{O}'$ and $\mathcal{A}'$ such that $\mathcal{I} \not\models q(a)$. Since $\mathcal{O}' \models \mathcal{O}$, $\mathcal{I}$ is also a model of $\mathcal{O}$. Since $\mathcal{O}' \models X_{\exists R.C} \sqsubseteq \exists R.C$ for all $\exists R.C \in \mathfrak{C}(\mathcal{O})$ and due to the construction of $\mathcal{A}$ from $\mathcal{A}'$, $\mathcal{I}$ is also a model of $\mathcal{A}$. Thus, $\mathcal{I}$ witnesses $\mathcal{A}, \mathcal{O} \not\models q(a)$, as required.

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 $q_F \subseteq_{\mathcal{O}} q'$.

   Let q' be an ELIQ with $q \subseteq_{\mathcal{O}} q' \not\subseteq_{\mathcal{O}} q$. From $q \subseteq_{\mathcal{O}} q'$, it follows that q' does not use the fresh concept names $X_C(x)$ in $\mathcal{O}'$. Consequently, we obtain from Lemma 12 that $q \subseteq_{\mathcal{O}'} q' \not\subseteq_{\mathcal{O}'} q$. There is thus a $q_F^0(x) \in \mathcal{F}'$ with $q_F^0 \subseteq_{\mathcal{O}'} q'$. Assume that $q_F \in \mathcal{F}$ was obtained from q_F^0 in the construction of $\mathcal{F}$. It suffices to show that $q_F \subseteq_{\mathcal{O}} q'$. Assume to the contrary that this is not the case. Then there is an ABox $\mathcal{A}$ and an individual $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A}, \mathcal{O} \models q_F(a)$, but $\mathcal{A}, \mathcal{O} \not\models q'(a)$. We can proceed as in Point 1 above to show that $\mathcal{A}, \mathcal{O}' \models q_F(a)$ and $\mathcal{A}, \mathcal{O}' \not\models q'(a)$, in contradiction to $q_F^0 \subseteq_{\mathcal{O}'} q'$.

It remains to prove the second part of the Lemma: every $\mathcal{O}$-minimal ELIQ q is also $\mathcal{O}'$-minimal. Suppose that q is not $\mathcal{O}'$-minimal, that is, there is a variable x such that $q \equiv_{\mathcal{O}'} q'$ where $q' = q|_{\mathsf{var}(q) \setminus \{x\}}$. Clearly, we have $q \subseteq_{\mathcal{O}} q'$ as $q' \subseteq q$. Moreover, since $q' \subseteq_{\mathcal{O}'} q$ and Point 1 of Lemma 12, we have $q' \subseteq_{\mathcal{O}} q$. Hence, $q \equiv_{\mathcal{O}} q'$ and q is not $\mathcal{O}$-minimal. ❑

Lemma 2. *$\mathcal{F}$ is a frontier of $q(x_0)$ w.r.t. $\mathcal{O}$.*

Proof. We show that $\mathcal{F}$ fulfills the three conditions of frontiers. For Condition 1, let $p(x_0)$ be a query from $\mathcal{F}$. Then, since q is satisfiable w.r.t. $\mathcal{O}$, so is p. Hence it suffices to show $p(x_0) \to (\mathcal{U}_{q,\mathcal{O}}, x_0)$ by Lemma 9.

We extend the mapping $\cdot^{\downarrow}$ to be defined on all variables of p by considering the yet unmapped variables added in Step 2A of the construction. Let z be such a fresh variable added for $x \in \mathsf{var}(p)$, roles R, S and concept name A. Then $x^{\downarrow} \rightsquigarrow^{R}_{q,\mathcal{O}} A$ and by construction of $\mathcal{U}_{q,\mathcal{O}}$, there is a trace $x^{\downarrow}RA \in \Delta^{\mathcal{U}_{q,\mathcal{O}}}$. Set $z^{\downarrow} = x^{\downarrow}RA$. Now $\cdot^{\downarrow}$ is defined on all variables of p and, by construction of p, it is a homomorphism from p to $\mathcal{U}_{q,\mathcal{O}}$ with $x_0^{\downarrow} = x_0$ as required.

We start the proof of the second condition of frontiers with the following claim:

Claim 1. $p \not\subseteq_{\mathcal{O}} q_x$ for all $x \in \mathsf{var}(q)$ and $p(x) \in \mathcal{F}_0(x)$.

Proof of Claim 1. We show the claim by induction on the codepth of x in q, matching the inductive construction of $\mathcal{F}_0$. In the induction start, x has codepth 0. Then, by definition of codepth, there is no $R(x, y) \in q$ that is directed away from x_0 and all $p \in \mathcal{F}_0(x)$ must be obtained by dropping a concept atom.

Let $p(x)$ be a query from $\mathcal{F}_0(x)$ that is obtained by dropping the concept atom $A(x) \in q$. Then, by choice of $A(x)$, there is no $B(x) \in p$ with $\mathcal{O} \models B \sqsubseteq A$ and no $R(x, x') \in p$ with $\mathcal{O} \models \exists R \sqsubseteq A$. Hence, $A(x) \in q_x$ and $A(x) \notin \mathcal{U}_{p,\mathcal{O}}$, therefore $p \not\subseteq_{\mathcal{O}} q_x$.

In the induction step, let x have codepth > 0, let $p(x)$ be a query from $\mathcal{F}_0(x)$ and assume that the claim holds for all variables with smaller codepth. Let $\cdot^{\downarrow}$ be the extension of the original $\cdot^{\downarrow}$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q_x,\mathcal{O}}$ that exists by Lemma 10. If p is obtained by dropping a concept atom, then the same argument as in the induction start yields $p \not\subseteq_{\mathcal{O}} q_x$. If p is obtained by generalizing a subquery attached to a role atom $R(x, y) \in q_x$, assume for contradiction that there is a homomorphism h from q_x to $\mathcal{U}_{p,\mathcal{O}}$ with $h(x) = x$. From h we construct a homomorphism h' from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h'(x_0) = x_0$ by setting $h'(z) = h(z)^{\downarrow}$ for all $z \in \mathsf{var}(q_y)$ and $h'(z) = z$ for all $z \notin \mathsf{var}(q_y)$. Since h must map y to an R-successor of x in $\mathcal{U}_{p,\mathcal{O}}$, we may distinguish the following cases.

- $h(y)$ is a $z \in \mathsf{var}(p)$ with $z^{\downarrow} \neq y$.

  Then, by definition of $\cdot^{\downarrow}$, $z^{\downarrow} = z$, and therefore, by construction of h', $h'(y) = h'(z) = z$. Hence h' is a non-injective homomorphism from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h'(x_0) = x_0$, contradicting $\mathcal{O}$-minimality or $\mathcal{O}$-saturatedness of q by Lemma 11.

- $h(y)$ is a trace $h(x)SA \in \Delta^{\mathcal{U}_{p,\mathcal{O}}}$ for some role S with $\mathcal{O} \models S \sqsubseteq R$ and concept name A.

  Then, if $h'(y)$ is also a trace, there must be a $y' \in \mathsf{var}(q)$ with $y' \notin \mathsf{img}(h')$, again contradicting $\mathcal{O}$-minimality or $\mathcal{O}$-saturatedness of q by Lemma 11.

  If $h'(y)$ is not a trace, but a successor y' of x with $y' \neq y$, then $h'(y') = h'(y)$, again contradicting $\mathcal{O}$-minimality of q by Lemma 11.

  If $h'(y) = y$ and there is a $y' \in \mathsf{var}(q_y)$ with $h'(y') = x$, then $h'(y') = h'(x)$, again contradicting $\mathcal{O}$-minimality by Lemma 11.

  If $h'(y) = y$ and there is no $y' \in \mathsf{var}(q_y)$ with $h'(y') = x$, then we show a contradiction to $\mathcal{O}$-minimality q by constructing a homomorphism h'' from q to $\mathcal{U}_{q',\mathcal{O}}$ with $h''(x_0) = x_0$ where q' is the restriction of q to $\mathsf{var}(q) \setminus \{y\}$. Note that by construction of h', $h'(z) = y$ implies $z = y$.

  Since $h(x)SA^{\downarrow} = y$, there is no trace $xSA \in \Delta^{\mathcal{U}_{q,\mathcal{O}}}$. But, since $h(x)SA \in \Delta^{\mathcal{U}_{p,\mathcal{O}}}$ it must be that $h(x) \leadsto^{S}_{p,\mathcal{O}} A$ and thus $\mathcal{A}_p, \mathcal{O} \models \exists R.A(h(x))$ and $\mathcal{A}_q, \mathcal{O} \models \exists R.A(x)$. However, $x \not\leadsto^{S}_{p,\mathcal{O}} A$ because $\mathcal{O} \models R \equiv S$, $R(x, y) \in q$ and $\mathcal{A}_q, \mathcal{O} \models A(y)$.

  Since $R(x, y) \notin q'$ and by $\mathcal{O}$-saturation of q and normal form of $\mathcal{O}$, it follows that $x \leadsto^{S}_{q',\mathcal{O}} A$ and therefore there is a trace $xSA \in \Delta^{\mathcal{U}_{q',\mathcal{O}}}$.

  Construct h'' by setting $h''(z) = h'(z)$ for all $z \in \mathsf{var}(q) \setminus \mathsf{var}(q_y)$ and $h''(z) = xSAR_2A_2 \dots R_nA_n$ for all $z \in \mathsf{var}(q_y)$ if $h(z) = xSAR_2A_2 \dots R_nA_n$.

- $h(y)$ is the root y' of a $p' \in \mathcal{F}_0(y)$ that was added in Point 3 of generalizing a subquery.

  Then, by the induction hypothesis, $q_y \not\to \mathcal{U}_{p',\mathcal{O}}$ for all $p' \in \mathcal{F}_0(y)$. We argue that, consequently, h cannot map q_y entirely into the subtree below y' in $\mathcal{U}_{p,\mathcal{O}}$. To show this, it clearly suffices to argue that this subtree is isomorphic to $\mathcal{U}_{p',\mathcal{O}}$. In fact, this is the case by definition of universal models, $\mathcal{O}$-saturation of p and since $\mathcal{O}$ is in normal form, unless there is a concept name A with $\mathcal{O} \models \exists R^- \sqsubseteq A$ such that $A(y') \in \mathcal{U}_{p,\mathcal{O}}$, but $A(y') \notin \mathcal{U}_{p',\mathcal{O}}$. Because of $\mathcal{O}$-saturatedness, this implies $A(y') \in p$ and $A(y') \notin p'$. Since the construction of $\mathcal{F}_0$ never adds concept names to an ELIQ, this implies that $A(y'^{\downarrow}) \in q$. Thus, $A(y'^{\downarrow})$ was dropped during the construction of p' from $q_{y'^{\downarrow}}$. This may only happen by dropping a concept atom. However, $R(x, y) \in q$ and $\mathcal{O} \models \exists R^- \sqsubseteq A$ contradicts Condition (b) of dropping a concept atom .

  We have thus shown that h cannot map q_y entirely into the subtree below $y' \in \mathcal{U}_{p',\mathcal{O}}$. Consequently, there must be a $y'' \in \mathsf{var}(q_y)$ with $h(y'') = x$. This yields $h'(y') = h'(x) = x$ contradicting $\mathcal{O}$-minimality of q by Lemma 11.

Note that $h(y)$ cannot be a variable introduced in Point 4 of generalizing a subquery, as there only S-successors of x are introduced that satisfy $\mathcal{O} \not\models S \sqsubseteq R$. This completes the proof of Claim 1.

We continue by using Claim 1 to show that $p \not\subseteq_{\mathcal{O}} q$ for all $p \in \mathcal{F}$. Let p be a query from $\mathcal{F}$ and assume for contradiction that $p \subseteq_{\mathcal{O}} q$. Then, there is a homomorphism h from q to $\mathcal{U}_{p,\mathcal{O}}$ with $h(x_0) = x_0$. Let $\cdot^{\downarrow}$ be the extension of the original $\cdot^{\downarrow}$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q,\mathcal{O}}$ which exists by Lemma 10. We compose h and $\cdot^{\downarrow}$ to construct a homomorphism h' from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h'(x_0) = x_0$. By Claim 1, there is no homomorphism that maps q entirely into $\mathcal{U}_{p',\mathcal{O}}$ for any $p' \in \mathcal{F}_0(x_0)$. Hence, there must be an $x \in \mathsf{var}(q)$ such that $h(x)$ is a fresh variable added in the compensation step. By definition of that step and since q is connected, we may distinguish the following two cases:

- $h(x)$ is a fresh variable added in Step 2A. Then by definition of $\cdot^{\downarrow}$, $h'(x)$ is a trace, contradicting $\mathcal{O}$-minimality of q by Lemma 11.

- $h(x)$ is a fresh variable z added in Step 2B for the role atom $S(y, y') \in p$ with $z^{\downarrow} = y^{\downarrow}$.

  Then, since q is connected, there must be a predecessor x' of x with $h(x') = y$. Hence $h'(x) = h'(x') = y^{\downarrow}$, contradicting $\mathcal{O}$-minimality of q by Lemma 11.

This completes the proof of Condition 2 of frontiers.

It remains to show that Condition 3 of frontiers is satisfied. Let $q'(x_0)$ be an ELIQ that is satisfiable w.r.t. $\mathcal{O}$ such that $q \subseteq_{\mathcal{O}} q' \not\subseteq_{\mathcal{O}} q$. We may assume w.l.o.g. that q' is $\mathcal{O}$-saturated.

There is a homomorphism g from q' to $\mathcal{U}_{q,\mathcal{O}}$ with $g(x_0) = x_0$. We have to show that there is a $p \in \mathcal{F}$ with $p \subseteq_{\mathcal{O}} q'$. To do this, we construct in three steps a homomorphism h from q' to $\mathcal{U}_{p,\mathcal{O}}$ with $h(x_0) = x_0$ for some $p \in \mathcal{F}$. During all steps, we maintain the invariant

$$h(z)^{\downarrow} = g(z) \qquad (*)$$

for all variables $z \in \mathsf{var}(q')$ with $h(z)$ defined and $\cdot^{\downarrow}$ the extension of the original $\cdot^{\downarrow}$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q,\mathcal{O}}$. In the first step of the construction, we define h for an initial segment of q'.

Let $U \subseteq \mathsf{var}(q')$ be the smallest set of variables (w.r.t. $\subseteq$) of q' such that $x_0 \in U$ and $R(x, y) \in q'$ with $x \in U$, $g(y) \in \mathsf{var}(q)$, and $S(g(x), g(y)) \in q$ directed away from x_0 implies $y \in U$. Let q^U be the restriction of q' to the variables in U.

Claim 2. For all $x \in U$ with $q^U_x \not\subseteq_{\mathcal{O}} q_{g(x)}$, there is a $p \in \mathcal{F}_0(g(x))$ and a homomorphism h' from q^U_x to $\mathcal{U}_{p,\mathcal{O}}$ that satisfies $(*)$.

Proof of Claim 2. Let $y = g(x)$. We show Claim 2 by induction on the codepth of x in q^U. In the induction start, x has codepth 0. We distinguish the following cases:

- There is an $R(y, y') \in q_y$.

  Then let $p \in \mathcal{F}_0(y)$ be constructed by generalizing the subquery attached to the role atom $R(y, y')$ and set $h'(x) = y$. Since q is $\mathcal{O}$-saturated, $A(y) \in \mathcal{U}_{q,\mathcal{O}}$ implies $A(y) \in \mathcal{U}_{p,\mathcal{O}}$. This and $y = g(x)$ implies that h' is a homomorphism.

- There is no $R(y, y') \in q_y$.

  Then $q^U_x \not\subseteq_{\mathcal{O}} q_y$ implies that there is an $A(y) \in q_y$ with $A(x) \notin \mathcal{U}_{q^U_x,\mathcal{O}}$, and we must even find an A with these properties such that there is no $B(y) \in q_y$ with $\mathcal{O} \models B \sqsubseteq A$ and $\mathcal{O} \not\models A \sqsubseteq B$. This implies that Property (a) of dropping concept atoms is satisfied. Property (b) is satisfied since there is no $R(y, y') \in q_y$ and thus we may construct $p \in \mathcal{F}_0(y)$ by dropping the concept atom $A(y)$. Set $h'(x) = y$.

In the induction step, let x have codepth > 0 in q^U and assume that the claim holds for all variables of smaller codepth. From $q^U_x \not\subseteq_{\mathcal{O}} q_y$, it follows that $q_y \not\to (\mathcal{U}_{q^U_x,\mathcal{O}}, x)$. We distinguish the following cases:

- There is an $R(y, y') \in q_y$ such that $q_{y'} \not\to (\mathcal{U}_{q^U_{x'},\mathcal{O}}, x')$ for all $S(x, x') \in q^U_x$ with $\mathcal{O} \models S \sqsubseteq R$.

  Let $p \in \mathcal{F}_0(y)$ be constructed by generalizing the subquery attached to the role atom $R(y, y')$. We construct the homomorphism h' from q^U_x to $\mathcal{U}_{p,\mathcal{O}}$ by starting with $h'(x) = y$ and continuing to map all successors of x. Let $S(x, x') \in q^U_x$.

  If $g(x') \neq y'$, then extend h' to the subtree below x' by setting $h'(z) = g(z)$ for all $z \in \mathsf{var}(q_{x'})$.

  If $g(x') = y'$ and $\mathcal{O} \models S \equiv R$, then, by the induction hypothesis, there is a $p' \in \mathcal{F}_0(y')$ and a homomorphism h'' from $q_{x'}$ to $\mathcal{U}_{p',\mathcal{O}}$ with $h''(x') = y'$. Extend h' to the variables in $q'_{x'}$ by mapping $q'_{x'}$ according to h'' to the copy of p' that was attached to y in Point 3 of generalizing a subquery.

  If $g(x') = y'$ and $\mathcal{O} \not\models S \sqsubseteq R$, then extend h' to the variables in $q'_{x'}$ by mapping $q'_{x'}$ according to g to the copy of $q_{y'}$ that was added in Point 4 of generalizing the subquery attached with the role S.

- For every $R(y, y') \in q_y$, $q_{y'} \to (\mathcal{U}_{q^U_{x'},\mathcal{O}}, x')$ for some $S(x, x') \in q^U_x$ with $\mathcal{O} \models S \sqsubseteq R$.

  Then there is an $A(y) \in q_y$ with $A(x) \notin \mathcal{U}_{q^U_x,\mathcal{O}}$ and we must even find an A with these properties such that there is no $B(y) \in q_y$ with $\mathcal{O} \models B \sqsubseteq A$ and $\mathcal{O} \not\models A \sqsubseteq B$. Thus, Property (a) of dropping concept atoms is satisfied. To show that Property (b) is also satisfied, we have to argue that there is no $R(y, y') \in q_y$ with $\mathcal{O} \models \exists R \sqsubseteq A$. But if there is such an $R(y, y') \in q_y$, then $q_{y'} \to (\mathcal{U}_{q^U_{x'},\mathcal{O}}, x')$ for some $S(x, x') \in q^U_x$ with $\mathcal{O} \models S \sqsubseteq R$. This implies $A(x) \in \mathcal{U}_{q^U_x,\mathcal{O}}$, a contradiction.

  We may thus construct $p \in \mathcal{F}_0(y)$ by dropping the concept atom $A(y)$. Set $h'(x') = g(x')$ for all $x' \in \mathsf{var}(q^U_x)$.

This completes the proof of Claim 2.

By Claim 2, there is a $p' \in \mathcal{F}_0(x_0)$ such that $q^U \to \mathcal{U}_{p',\mathcal{O}}$. Let $p \in \mathcal{F}$ be the query that was obtained by applying the compensation step to p'. Then clearly also $q^U \to \mathcal{U}_{p,\mathcal{O}}$. Define h for all variables in U according to the homomorphism that witnesses this. Let $\cdot^{\downarrow}$ be the extension of the original $\cdot^{\downarrow}$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q,\mathcal{O}}$ which exists by Lemma 10.

We continue with the second step of the construction of h which covers subtrees of q' that are connected to the initial segment q^U and whose root is mapped by g to traces of $\mathcal{U}_{q,\mathcal{O}}$ (and as opposed to a variable from $\mathsf{var}(q)$). Consider all atoms $R(x, x') \in q'$ with $h(x)$ defined, $h(x')$ undefined and $g(x') \notin \mathsf{var}(q)$. Before extending h to $q'_{x'}$, we first show that there is an atom $S(h(x), z) \in p$ with $\mathcal{O} \models S \sqsubseteq R$, added in Step 2A.

Since $g(x') \notin \mathsf{var}(q)$, $g(x')$ must be a trace $g(x)SA \in \Delta^{\mathcal{U}_{q,\mathcal{O}}}$ for some concept name A and role S with $\mathcal{O} \models S \sqsubseteq R$. Hence, $g(x) \rightsquigarrow^S_{q,\mathcal{O}} A$.

We aim to show that Step 2A of compensation is applicable. To thus end, take any concept name B such that $\mathcal{O} \models \exists R \sqsubseteq B$. We have to show that $B(h(x)) \in p$. Assume to the contrary that $B(h(x)) \notin p$. Then, since q is $\mathcal{O}$-saturated, $B(g(x)) \in q$ and p must be the result of dropping the concept atom $B(g(x))$. However, since q' is $\mathcal{O}$-saturated and $R(x, x') \in q'$, $B(x) \in q'$ and this contradicts Claim 2, since $x \in U$.

Hence, Step 2A adds the atoms $R(h(x), z), B(z), S(z', z)$ with z and z' fresh variables and adds a disjoint copy $\widehat{q}$ of q, gluing the copy of $h(x)^{\downarrow}$ in $\widehat{q}$ to z'. Extend h to the variables in $q'_{x'}$ by setting $h(\hat{x}) = zR_2M_2 \dots R_nM_n$ if $g(\hat{x}) = g(x)SAR_2M_2 \dots R_nM_n$ for all $\hat{x}$ in the subtree below x'. If there is an $x'' \in \mathsf{var}(q'_{x'})$ with $g(x'') = g(x)$, set $h(x'') = z'$ and continue mapping the subtree below x'' into the attached copy $\widehat{q}$ of q according to g.

In the third and final step of the construction of h, we consider the remaining subtrees of q'. Let $R(x, x') \in q'$ be directed away from x_0 with $h(x)$ defined and $h(x')$ undefined.

Then $h(x)$ was defined in the first step of the construction of h, and thus $x \in U$. As $h(x')$ was not defined in the second step $g(x') \in \mathsf{var}(q)$. Therefore, since $x' \notin U$, $R(g(x), g(x')) \in \mathcal{U}_{q,\mathcal{O}}$ must be directed towards x_0. This implies that x is not the root of q' and that there is an atom $T(x'', x) \in q'$ directed away from x_0 with $g(x'') = g(x')$. From $x \in U$ it follows that $x'' \in U$ and therefore $h(x'')$ and $h(x)$ were defined in the first step of the construction of h.

Hence, there is an atom $S(h(x''), h(x)) \in p$ directed away from x_0 with $\mathcal{O} \models S \sqsubseteq T$ that was not added in Step 2A.

Since by $(*)$ $h(x'')^{\downarrow} = g(x'') = g(x')$ and $h(x)^{\downarrow} = g(x)$, $\mathcal{A}_q, \mathcal{O} \models R^-(h(x'')^{\downarrow}, h(x)^{\downarrow})$

Therefore, Step 2B added an atom $R^-(z, h(x))$ to p, z a fresh variable, and glued a copy $\widehat{q}$ of q to z. Set $h(x') = z$ and extend h to the entire subtree $q'_{x'}$ by mapping all variables into the attached copy of q according to g.

This completes the construction of h and the proof that Condition 3 is satisfied. ❏

Lemma 3. *The construction of $\mathcal{F}$ runs in time polynomial in $||q|| + ||\mathcal{O}||$ (and thus $\sum_{p\in\mathcal{F}} ||p||$ is polynomial in $||q|| + ||\mathcal{O}||$).*

Proof. In order to reduce notational clutter, we introduce some abbreviations used throughout the proof.

- $s = |\mathsf{sig}(q)|$ denotes the number of concept and role names used in q;
- $o = ||\mathcal{O}||$ denotes the size of $\mathcal{O}$;
- for an ELIQ p, $n_p = |\mathsf{var}(p)|$ denotes the number of variables in p;
- for a set Q of queries, n_Q denotes $\sum_{p\in Q} n_p$.

We assume without loss of generality that s and o are at least one.

We start with analyzing the size of the queries in $\mathcal{F}_0(x)$ that are obtained as the result of the generalization step.

Claim. For every $x \in \mathsf{var}(q)$,

$$n_{\mathcal{F}_0(x)} \leq s \cdot o \cdot n_{q_x}^3.$$

Proof of the claim. The proof is by induction on the codepth of x in q. For the base case, consider a variable x of codepth 0 in q, that is, a leaf. In this case, only Step (A) is applicable, and it adds at most s queries to $\mathcal{F}_0(x)$, each with a single variable.

For the inductive step, consider a variable x of codepth greater than 0. We partition $\mathcal{F}_0(x)$ into $\mathcal{F}_0^A(x)$ and $\mathcal{F}_0^B(x)$, that is, the queries that are obtained by dropping a concept atom in Step (A) and the queries that are obtained by generalizing a subquery in Step (B), respectively, and analyze them separately, starting with $\mathcal{F}_0^A(x)$. Clearly, every $p \in \mathcal{F}_0^A(x)$ uses n_{q_x} variables and there are at most s queries in $\mathcal{F}_0^A$. Thus, we have

$$n_{\mathcal{F}_0^A(x)} \leq s \cdot n_{q_x}.$$

Next, we analyze $\mathcal{F}_0^B(x)$. Each query in $\mathcal{F}_0^B(x)$ is obtained by first picking, in Point 1, an atom $R(x, y)$ in q_x. Then, in Point 3, we add $\sum_{p\in\mathcal{F}_0(y)} n_p$ variables and, in Point 4, we add at most o copies of q_y. Thus, we obtain

$$n_{\mathcal{F}_0^B(x)} \leq \sum_{R(x,y)\in q_x} (n_{q_x} + n_{\mathcal{F}_0(y)} + o \cdot n_{q_y}).$$

Plugging in the induction hypothesis, we obtain

$$n_{\mathcal{F}_0^B(x)} \leq \sum_{R(x,y)\in q_x} (n_{q_x} + s \cdot o \cdot n_{q_y}^3 + o \cdot n_{q_y}). \tag{1}$$

We simplify the right-hand side of (1) by making the following observations:

- $\sum_{R(x,y)\in q_x} n_{q_x} \leq n_{q_x} \cdot (n_{q_x} - 1)$,
- $\sum_{R(x,y)\in q_x} n_{q_y} = n_{q_x} - 1$, and
- $\sum_{R(x,y)\in q_x} n_{q_y}^3 \leq \left(\sum_{R(x,y)\in q_x} n_{q_y}\right)^3 = (n_{q_x} - 1)^3$. Here, the inequality is an application of the general inequality $\sum_i a_i^3 \leq (\sum_i a_i)^3$, for every sequence of non-negative numbers $a_1, \ldots, a_k$.

Using these observations, Inequality (1) can be simplified to:

$$\begin{aligned} n_{\mathcal{F}_0^B(x)} &\leq n_{q_x} \cdot (n_{q_x} - 1) + s \cdot o \cdot (n_{q_x} - 1)^3 + o \cdot (n_{q_x} - 1) \\ &\leq s \cdot o \cdot \left(n_{q_x} \cdot (n_{q_x} - 1) + (n_{q_x} - 1)^3 + (n_{q_x} - 1)\right) \\ &= s \cdot o \cdot \left(n_{q_x}^2 + (n_{q_x} - 1)^3 - 1\right). \end{aligned}$$

Overall, we get

$$\begin{aligned} n_{\mathcal{F}_0(x)} &= n_{\mathcal{F}_0^A(x)} + n_{\mathcal{F}_0^B(x)} \\ &\leq s \cdot n_{q_x} + s \cdot o \cdot \left(n_{q_x}^2 + (n_{q_x} - 1)^3 - 1\right) \\ &\leq s \cdot o \cdot \left(n_{q_x} + n_{q_x}^2 + (n_{q_x} - 1)^3 - 1\right) \\ &\leq s \cdot o \cdot n_{q_x}^3. \end{aligned}$$

In the last inequality, we used that $z^3 \geq z + z^2 + (z-1)^3 - 1$, for all real numbers z. This finishes the proof of the claim.

We analyze now the compensation Step 2, in which the queries in $\mathcal{F}_0(x_0)$ are further extended. We denote with $\mathcal{F}_1$ the result of applying Step 2A to $\mathcal{F}_0(x_0)$. In Step 2A, we add at most one variable and a copy of q for every variable in $\mathcal{F}_0(x_0)$ and every choice of a concept name A and role names S and R that occurs in $\mathcal{O}$. Therefore, we add at most $(1 + n_q) \cdot n_{\mathcal{F}_0(x_0)} \cdot o^3$ variables in total. Using the claim, we get

$$\begin{aligned} n_{\mathcal{F}_1} &\leq n_{\mathcal{F}_0(x_0)} + (1 + n_q) \cdot n_{\mathcal{F}_0(x_0)} \cdot o^3 \\ &\leq s \cdot o \cdot n_q^3 \cdot \left(1 + (1 + n_q) \cdot o^3\right). \end{aligned}$$

In Step 2B, we add at most one copy of q for every role atom in some query in $\mathcal{F}_1$ and every role name in $\mathcal{O}$. Hence,

$$\begin{aligned} n_{\mathcal{F}} &\leq n_{\mathcal{F}_1} + n_{\mathcal{F}_1} \cdot n_q \cdot o \\ &= \left(s \cdot o \cdot n_q^3 \cdot \left(1 + (1 + n_q) \cdot o^3\right)\right) \cdot (1 + n_q \cdot o), \end{aligned}$$

which is polynomial in $||q||$ and $||\mathcal{O}||$. Moreover, the computation of $\mathcal{F}$ can be carried out in polynomial time since all the involved queries are of polynomial size and consequences of $\mathcal{O}$ can be decided in polynomial time. ❏

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

$$q_n(x) = A_1(x) \wedge A'_1(x) \wedge \cdots \wedge A_n(x) \wedge A'_n(x)$$

$$\mathcal{O}_n = \{A_i \sqcap A'_i \sqsubseteq A_1 \sqcap A'_1 \sqcap \cdots \sqcap A_n \sqcap A'_n \mid 1 \leq i \leq n\}.$$

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.

❑

D Proofs for Section 4

A *CQ-frontier* for an ELIQ q w.r.t. $\mathcal{O}$ is a finite set of unary CQs that satisfies Properties 1–3 of Definition 1. Note that every frontier is a CQ-frontier, but not vice versa.

Theorem 3. *There is an ELIQ q and a DL-Lite$^{\mathcal{F}}$ ontology $\mathcal{O}$ such that q does not have a finite frontier w.r.t. $\mathcal{O}$.*

Proof. Let $q(x) = A(x)$ and

$$\mathcal{O} = \{\ A \sqsubseteq \exists r, \quad \exists r^- \sqsubseteq \exists r, \quad \exists r \sqsubseteq \exists s, \quad \mathsf{func}(r^-)\ \}.$$

The universal model $\mathcal{U}_{q,\mathcal{O}}$ of $\mathcal{A}_q$ and $\mathcal{O}$ is an infinite r-path in which every point has a single s-successor.

Suppose, for the sake of showing a contradiction, that $\mathcal{F}$ is a CQ-frontier of q w.r.t. $\mathcal{O}$. We can assume w.l.o.g. that all queries in $\mathcal{F}$ are satisfiable w.r.t. $\mathcal{O}$, especially that they satisfy $\mathsf{func}(r^-)$. Since $\mathcal{F}$ is finite, there is an $n \geq 1$ such that $|\mathsf{var}(p)| < n$, for all $p \in \mathcal{F}$. Consider the following ELIQ q':

$$\begin{aligned} q'(x_1) = {} & r(x_1, x_2), \ldots, r(x_{n-1}, x_n), \\ & s(x_n, y), s(x'_n, y), \\ & r(x'_1, x'_2), \ldots, r(x'_{n-1}, x'_n), A(x'_1). \end{aligned}$$

Note that $q' \not\subseteq_{\mathcal{O}} q \subseteq_{\mathcal{O}} q'$ and that q' satisfies $\mathsf{func}(r^-)$.

By Property 3 of frontiers, there is a query $p(z) \in \mathcal{F}$ such that $p \subseteq_{\mathcal{O}} q'$. By Lemma 9, there is a homomorphism h from q' to $\mathcal{U}_{p,\mathcal{O}}$ with $h(x_1) = z$. We distinguish cases.

Suppose first that $h(x_i) \in \mathsf{var}(p)$ for all i with $1 \leq i \leq n$, then by the choice of n there must be $1 \leq i < j \leq n$ such that $h(x_i) = h(x_j)$. Since q' contains a directed r-path from x_i to x_j and $\mathcal{U}_{p,\mathcal{O}}$ does not contain edges between variables that are not part of p, this implies that p must contain an r-cycle. Thus, $q \not\subseteq_{\mathcal{O}} p$, violating Property 1 of frontiers.

Suppose now that $h(x_i) \notin \mathsf{var}(p)$ for some i with $1 \leq i \leq n$, that is, $h(x_i)$ is a trace starting with some $y \in \mathsf{var}(p)$. Since q' is an ELIQ, there is a $j < i$ such that $h(x_j) = y$ and $h(x_{j+1}), \ldots, h(x_i) \notin \mathsf{var}(p)$. The structure of q' and the structure of the anonymous part in universal models of $\mathcal{O}$ imply that $h(x'_j) = h(x_j)$.

We now show that $h(x_1) = h(x'_1)$. If $j = 1$, we are done. If $j > 1$, there are atoms $r(x_{j-1}, x_j)$ and $r(x'_{j-1}, x'_j)$ in q'. Since h is a homomorphism, $h(x_j) = h(x'_j)$, and p satisfies $\mathsf{func}(r^-)$, we obtain $h(x_{j-1}) = h(x'_{j-1})$. Repeating this argument yields $h(x_1) = h(x'_1)$ as required. Since $h(x_1) = z$, we also have $h(x'_1) = z$. Since h is a homomorphism and $A(x'_1) \in q'$, we have $A(z) \in p$ and thus $p \subseteq_{\mathcal{O}} q$, violating Property 2 of frontiers. ❑

Lemma 4. *For every DL-Lite$^{\mathcal{F}-}$ ontology $\mathcal{O}$, we can construct in polynomial time a DL-Lite$^{\mathcal{F}-}$ 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. The proof of the Lemma is the same as the proof of Lemma 1, except for the verification that the constructed set $\mathcal{F}$ satisfies Property 2 of frontiers. So we detail this here.

Claim. $q_F \not\subseteq q$, for all $q_F \in \mathcal{F}$.

Proof of the claim. Let $q_F(x) \in \mathcal{F}$ and let $q_F^0(x) \in \mathcal{F}'$ be the ELIQ from which $q_F(x)$ was obtained during the construction of $\mathcal{F}$. Since $q_F^0 \not\subseteq_{\mathcal{O}'} q$, there is an ABox $\mathcal{A}'$ and an individual $a \in \mathsf{ind}(\mathcal{A}')$ such that $\mathcal{A}', \mathcal{O}' \models q_0^F(a)$, but $\mathcal{A}', \mathcal{O}' \not\models q(a)$. As in the proof of Lemma 1, the idea is to obtain an ABox $\mathcal{A}$ by starting with $\mathcal{A}'$ and adding $\mathcal{A}_{C(b)}$, for each concept assertion $X_C(b) \in \mathcal{A}'$. However, the addition of $\mathcal{A}_{C(b)}$, with C an $\mathcal{ELI}$-concept has to respect the functionality assertions of $\mathcal{O}$. In order to achieve this, we define the *addition of a tree-shaped ABox $\mathcal{B}$ with root b_0 to $\mathcal{A}$ at a* inductively on the structure of $\mathcal{B}$ as follows:

1. for all $A(b_0) \in \mathcal{B}$, add $A(a)$ to $\mathcal{A}$;
2. for all $R(b_0, b') \in \mathcal{B}$, let $\mathcal{B}'$ be the sub-ABox of $\mathcal{B}$ rooted at b' and
   (a) if $\mathsf{func}(R) \in \mathcal{O}$ and there is an atom $R(a, a')$ in $\mathcal{A}$, then add $\mathcal{B}'$ to $\mathcal{A}$ at a';
   (b) otherwise, add an atom $R(a, a')$ for a fresh individual a' and add $\mathcal{B}'$ to $\mathcal{A}$ at a'.

Importantly, there can only be one atom $R(b, b')$ in the first case for the existential restriction, if $\mathcal{A}$ satisfies the functionality assertions in $\mathcal{O}$. It should be clear that the resulting ABox also satisfies the functionality assertions if $\mathcal{A}$ does.

We thus obtain the ABox $\mathcal{A}$ by starting with $\mathcal{A} = \mathcal{A}'$ and adding $\mathcal{A}_{C(b)}$ to $\mathcal{A}$ at b, for each concept assertion $X_C(b) \in \mathcal{A}'$. We aim to show $\mathcal{A}, \mathcal{O} \models q_F(a)$ and $\mathcal{A}, \mathcal{O} \not\models q(a)$, witnessing $q_F \not\subseteq_{\mathcal{O}} q$ as required.

For the former, assume to the contrary that $\mathcal{A}, \mathcal{O} \not\models q_F(a)$. Then there is a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{O}$ with $\mathcal{I} \not\models q_F(a)$. Let $\mathcal{I}'$ be the extension of $\mathcal{I}$ according to Point 2 of Lemma 12. Then $\mathcal{I}'$ is a model of $\mathcal{O}'$ and, by construction of $\mathcal{I}'$ and of $\mathcal{A}$ from $\mathcal{A}'$, also a model of $\mathcal{A}'$. Moreover, $\mathcal{I} \not\models q_F(a)$ implies $\mathcal{I}' \not\models q_F^0(a)$ by construction of q_F and of $\mathcal{I}'$. This contradicts $\mathcal{A}', \mathcal{O}' \models q_0^F(a)$.

It remains to show that $\mathcal{A}, \mathcal{O} \not\models q(a)$. Since $\mathcal{A}', \mathcal{O}' \not\models q(a)$, there is a model $\mathcal{I}$ of $\mathcal{O}'$ and $\mathcal{A}'$ such that $\mathcal{I} \not\models q(a)$. Since $\mathcal{O}' \models \mathcal{O}, \mathcal{I}$ is also a model of $\mathcal{O}$. Since $\mathcal{O}' \models X_{\exists R.C} \sqsubseteq \exists R.C$ for all $\exists R.C \in \mathfrak{C}(\mathcal{O})$ and due to the construction of $\mathcal{A}$ from $\mathcal{A}'$, $\mathcal{I}$ is also a model of $\mathcal{A}$. Thus, $\mathcal{I}$ witnesses $\mathcal{A}, \mathcal{O} \not\models q(a)$, as required.

This finishes the proof of the claim and of the lemma.

❑

Lemma 5. *$\mathcal{F}$ is a frontier of $q(x_0)$ w.r.t. $\mathcal{O}$.*

Proof. We show that $\mathcal{F}$ fulfills the three conditions of frontiers. For Condition 1, let $p(x_0)$ be a query from $\mathcal{F}$ and let $p_0(x_0) \in \mathcal{F}_0(x_0)$ be the query that was used to construct p

by applying the compensation step. First we observe that p_0 is satisfiable w.r.t. $\mathcal{O}$, since q is satisfiable w.r.t. $\mathcal{O}$ and neither the dropping of a concept atom nor the generalizing of a subquery introduces any violations of functionality assertions in $\mathcal{O}$. Next, let $R(x,z) \in p$ be an atom that was added during Step 2A of the compensation step. Then either $\mathsf{func}(R) \notin \mathcal{O}$ or there is no atom $R(x,z') \in p$ and therefore Step 2A introduces no atoms that violate any functionality assertions. Similarly Step 2B ensures that all added atoms to not violate functionality assertions in $\mathcal{O}$. Therefore p is also satisfiable w.r.t. $\mathcal{O}$. Hence it suffices to show $p(x_0) \to (\mathcal{U}_{q,\mathcal{O}}, x_0)$ by Lemma 9.

We extend the mapping $\cdot^{\downarrow}$ to be defined on all variables of p by considering the yet unmapped variables added in Step 2A and Point (iii) of Step 2B. Let u be a fresh variable with $u^{\downarrow}$ undefined that was added because there is a $x \in \mathsf{var}(p)$, a role R and a set M of concept names such that $x^{\downarrow} \leadsto^{R}_{q,\mathcal{O}} M$. Then, by construction of $\mathcal{U}_{q,\mathcal{O}}$ there is a trace $x^{\downarrow}RM \in \Delta^{\mathcal{U}_{q,\mathcal{O}}}$. Set $u^{\downarrow} = x^{\downarrow}RM$. Now $\cdot^{\downarrow}$ is defined on all variables of p and, by construction of p, it is a homomorphism from p to $\mathcal{U}_{q,\mathcal{O}}$ with $x_0^{\downarrow} = x_0$ as required.

We start the proof of the second condition of frontiers with the following claim:

Claim 1. $p \not\subseteq_{\mathcal{O}} q_x$ for all $x \in \mathsf{var}(q)$ and $p(x) \in \mathcal{F}_0(x)$.

Proof of Claim 1. We show the claim by induction on the codepth of x in q, matching the inductive construction of $\mathcal{F}_0$. In the induction start, x has codepth 0. Then, by definition of codepth, there is no $R(x,y) \in q$ that is directed away from x_0 and all $p \in \mathcal{F}_0(x)$ are obtained by dropping a concept atom.

Let $p(x)$ be a query from $\mathcal{F}_0(x)$ that is obtained by dropping the concept atom $A(x) \in q$. Then, by choice of $A(x)$, there is no $B(x) \in p$ with $\mathcal{O} \models B \sqsubseteq A$ and no $R(x,x') \in p$ with $\mathcal{O} \models \exists R \sqsubseteq A$. Hence $A(x) \in q_x$, but $A(x) \notin \mathcal{U}_{p,\mathcal{O}}$ and therefore $p \not\subseteq_{\mathcal{O}} q_x$.

In the induction step, let x have codepth > 0, let $p(x)$ be a query from $\mathcal{F}_0(x)$ and assume that the claim holds for all variables with smaller codepth. Let $\cdot^{\downarrow}$ be the extension of the original $\cdot^{\downarrow}$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q_x,\mathcal{O}}$, which exists by Lemma 10. If p is obtained by dropping a concept atom, then the same argument as in the induction start yields $p \not\subseteq_{\mathcal{O}} q_x$. If p is obtained by generalizing the subquery attached to a role atom $R(x,y) \in q$, assume for contradiction that there is a homomorphism h from q_x to $\mathcal{U}_{p,\mathcal{O}}$ with $h(x) = x$. From h we construct a homomorphism h' from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h'(x_0) = x_0$ by setting $h'(z) = h(z)^{\downarrow}$ for all $z \in \mathsf{var}(q_y)$ and $h'(z) = z$ for all $z \notin \mathsf{var}(q_y)$. Since h must map y to a R-successor of x in $\mathcal{U}_{p,\mathcal{O}}$, we may distinguish the following cases.

- $h(y)$ is a variable $z \in \mathsf{var}(p)$ with $z^{\downarrow} \neq y$. Then, by definition of $\cdot^{\downarrow}$, $z^{\downarrow} = z$, and therefore, by construction of h', $h'(y) = h'(z) = z$. Hence, h' is a non-injective homomorphism from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h'(x_0) = x_0$, contradicting $\mathcal{O}$-minimality or $\mathcal{O}$-saturatedness of q by Lemma 11.
- $h(y)$ is a trace $h(x)RM \in \mathcal{U}_{p,\mathcal{O}}$ for some set M of concept names.

  If $h'(y)$ is also a trace, then there must be a $y' \in \mathsf{var}(q)$ with $y' \notin \mathsf{img}(h')$, again contradicting $\mathcal{O}$-minimality or $\mathcal{O}$-saturatedness of q by Lemma 11.

  If $h'(y)$ is not a trace, but a different successor y' of x, then $h'(y') = h'(y)$, again contradicting $\mathcal{O}$-minimality of q by Lemma 11.

  If $h'(y) = y$ and there is no $y' \in \mathsf{var}(q_y)$ with $h'(y') = x$, then we show a contradiction to $\mathcal{O}$-minimality q by constructing a homomorphism h'' from q to $\mathcal{U}_{q',\mathcal{O}}$ with $h''(x_0) = x_0$ where q' is the restriction of q to $\mathsf{var}(q) \setminus \{y\}$. Note that by construction of h', $h'(z) = y$ implies $z = y$.

  Since $h(x)RM^{\downarrow} = y$, there is no trace $xRM \in \Delta^{\mathcal{U}_{q,\mathcal{O}}}$. But, since $h(x)RM \in \Delta^{\mathcal{U}_{p,\mathcal{O}}}$ it must be that $h(x) \leadsto^{S}_{p,\mathcal{O}} M$ and thus $\mathcal{A}_p, \mathcal{O} \models \exists R.\bigsqcap M(h(x))$ and $\mathcal{A}_q, \mathcal{O} \models \exists R.\bigsqcap M(x)$. However, $x \not\leadsto^{S}_{p,\mathcal{O}} M$ because $R(x,y) \in q$ and $\mathcal{A}_q, \mathcal{O} \models \bigsqcap M(y)$.

  Since $R(x,y) \notin q'$ and by $\mathcal{O}$-saturation of q and normal form of $\mathcal{O}$, it follows that $x \leadsto^{R}_{q',\mathcal{O}} M$ and therefore there is a trace $xRM \in \Delta^{\mathcal{U}_{q',\mathcal{O}}}$.

  Construct h'' by setting $h''(z) = h'(z)$ for all $z \in \mathsf{var}(q) \setminus \mathsf{var}(q_y)$ and $h''(z) = xRMR_2M_2 \dots R_nM_n$ for all $z \in \mathsf{var}(q_y)$ if $h(z) = xRMR_2M_2 \dots R_nM_n$.
- $h(y)$ is the root y' of a $p' \in \mathcal{F}_0(y)$ that was added in Points 3 or 4 of generalizing a subquery.

  By the induction hypothesis, $q_y \not\to \mathcal{U}_{p',\mathcal{O}}$ for all $p' \in \mathcal{F}_0(y)$. We argue that, consequently, h cannot map q_y entirely into the subtree below y' in $\mathcal{U}_{p,\mathcal{O}}$. To show this, it clearly suffices to argue that this subtree is isomorphic to $\mathcal{U}_{p',\mathcal{O}}$. In fact, this is the case by definition of universal models and since $\mathcal{O}$ is in normal form, unless there is a concept name A with $\mathcal{O} \models \exists R^- \sqsubseteq A$ such that $A(y') \in \mathcal{U}_{p,\mathcal{O}}$, but $A(y') \notin \mathcal{U}_{p',\mathcal{O}}$. Because of $\mathcal{O}$-saturatedness, this implies $A(y') \in p$ and $A(y') \notin p'$. Since the construction of $\mathcal{F}_0$ never adds concept names to an ELIQ, this implies that $A(y'^{\downarrow}) \in q$. Thus, $A(y'^{\downarrow})$ was dropped during the construction of p' from $q_{y'^{\downarrow}}$. This may only happen by dropping a concept atom. However, $R(x,y) \in q$ and $\mathcal{O} \models \exists R^- \sqsubseteq A$ contradicts Condition (b) of dropping a concept atom.

  We have thus shown that h cannot map q_y entirely into the subtree below $y' \in \mathcal{U}_{p',\mathcal{O}}$. Consequently, there must be a $y'' \in \mathsf{var}(q_y)$ with $h(y'') = x$. This yields $h'(y') = h'(x) = x$ contradicting $\mathcal{O}$-minimality of q by Lemma 11.

This completes the proof of Claim 1.

We continue by using Claim 1 to show that $p \not\subseteq_{\mathcal{O}} q$ for all $p \in \mathcal{F}$. Let p be a query from $\mathcal{F}$ and assume for contradiction that $p \subseteq_{\mathcal{O}} q$. Then, there is a homomorphism h from q to $\mathcal{U}_{p,\mathcal{O}}$ with $h(x_0) = x_0$. Let $\cdot^{\downarrow}$ be the extension of the original $\cdot^{\downarrow}$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q,\mathcal{O}}$, which exists by Lemma 10. We compose h and $\cdot^{\downarrow}$ to construct a homomorphism h' from q to $\mathcal{U}_{q,\mathcal{O}}$ with $h'(x_0) = x_0$. By Claim 1, there is no homomorphism that maps q entirely into $\mathcal{U}_{p',\mathcal{O}}$ for any $p' \in \mathcal{F}_0(x_0)$. Hence, there must be an $x \in \mathsf{var}(q)$ such that $h(x)$ is a fresh variable added in the compensation step.

By definition of that step and since q is connected, we may distinguish the following cases:

- $h(x)$ is a fresh variable added in Step 2A.

  Then, by definition of $\cdot^{\downarrow}$, $h'(x)$ is a trace, contradicting $\mathcal{O}$-minimality of q by Lemma 11.

- $h(x)$ is a fresh variable z added in the start of Step 2B for the role atom $R(y, y') \in p$ with $z^{\downarrow} = y^{\downarrow}$.

  Then, since q is connected, there must be a predecessor x' of x with $h(x') = y$. Hence $h'(x) = h'(x') = y^{\downarrow}$, contradicting $\mathcal{O}$-minimality of q by Lemma 11.

- $h(x)$ is a fresh variable added in the iterated step of Step 2B.

  Then, since q is connected and the step only adds variables to the subtree below a marked atom , there must be a predecessor x' of x such that $h(x')$ is a fresh variable added in the start of Step 2B. This leads to the same contradiction as in the last case.

This completes the proof of Condition 2 of frontiers.

It remains to show that Condition 3 of frontiers is satisfied. Let $q'(x_0)$ be an ELIQ that is satisfiable w.r.t. $\mathcal{O}$ such that $q \subseteq_{\mathcal{O}} q' \not\subseteq_{\mathcal{O}} q$. We may assume w.l.o.g. that q' is $\mathcal{O}$-saturated and that it satisfies all functionality assertions in $\mathcal{O}$. If, in fact, q' contains atoms $R(x, y_1), R(x, y_2)$ with $y_1 \neq y_2$ and $\mathsf{func}(R) \in \mathcal{O}$, then we can identify y_1 and y_2, obtaining an ELIQ that is equivalent w.r.t. $\mathcal{O}$ to the original q'.

There is a homomorphism g from q' to $\mathcal{U}_{q,\mathcal{O}}$ with $g(x_0) = x_0$. We have to show that there is a $p \in \mathcal{F}$ with $p \subseteq_{\mathcal{O}} q'$. To do this, we construct in four steps a homomorphism h from q' to $\mathcal{U}_{p,\mathcal{O}}$ with $h(x_0) = x_0$ for some $p \in \mathcal{F}$. During all steps, we maintain the invariant

$$h(z)^{\downarrow} = g(z) \qquad (*)$$

for all variables $z \in \mathsf{var}(q')$ with $h(z)$ defined and $\cdot^{\downarrow}$ the extension of the original $\cdot^{\downarrow}$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q,\mathcal{O}}$. In the first step of the construction, we define h for an initial segment of q'.

Let $U \subseteq \mathsf{var}(q')$ be the smallest set of variables (w.r.t. $\subseteq$) of q' such that $x_0 \in U$ and, for all $x \in U$ and $R(x, y) \in q'$ directed away from x_0, we have: if $R(g(x), g(y))$ is an atom in q directed away from x_0, then $y \in U$. Intuitively, U induces the maximal initial segment of q' that is mapped in a 'direction-preserving' way. Let q^U be the restriction of q' to the variables in U.

Claim 2. For all $x \in U$ with $q_x^U \not\subseteq_{\mathcal{O}} q_{g(x)}$, there is a $p \in \mathcal{F}_0(g(x))$ and a homomorphism h' from q_x^U to $\mathcal{U}_{p,\mathcal{O}}$ that satisfies $(*)$.

Proof of Claim 2. Let $y = g(x)$. We show Claim 2 by induction on the codepth of x in q^U. In the induction start, x has codepth 0. We distinguish the following cases:

- There is an $R(y, y') \in q_y$.

  Then let $p \in \mathcal{F}_0(y)$ be constructed by generalizing the subquery attached to $R(y, y')$ and set $h'(x) = y$. Since q is $\mathcal{O}$-saturated, $A(y) \in \mathcal{U}_{q,\mathcal{O}}$ implies $A(y) \in \mathcal{U}_{p,\mathcal{O}}$. This and $y = g(x)$ implies that h' is a homomorphism.

- There is no $R(y, y') \in q_y$.

  Then $q_x^U \not\subseteq_{\mathcal{O}} q_y$ implies that there is an $A(y) \in q_y$ with $A(x) \notin \mathcal{U}_{q_x^U,\mathcal{O}}$, and we must even find an A with these properties such that there is no $B(y) \in q_y$ with $\mathcal{O} \models B \sqsubseteq A$ and $\mathcal{O} \not\models A \sqsubseteq B$. This implies that Property (a) of dropping concept atoms is satisfied. Property (b) is satisfied since there is no $R(y, y') \in q_y$ and thus we may construct $p \in \mathcal{F}_0(y)$ by dropping the concept atom $A(y)$. Set $h'(x) = y$.

In the induction step, let x have codepth > 0 and assume that the claim holds for all variables of smaller codepth. From $q_x^U \not\subseteq_{\mathcal{O}} q_y$, it follows that $q_y \not\rightarrow (\mathcal{U}_{q_x^U,\mathcal{O}}, x)$. We distinguish the following cases:

- There is an $R(y, y') \in q_y$ such that $q_{y'} \not\rightarrow (\mathcal{U}_{q_{x'}^U,\mathcal{O}}, x')$ for all $R(x, x') \in q_x^U$.

  First assume $\mathsf{func}(R) \notin \mathcal{O}$. Then let $p \in \mathcal{F}_0(y)$ be constructed by generalizing the subquery attached to $R(y, y')$. We construct the homomorphism h' from q_x^U to $\mathcal{U}_{p,\mathcal{O}}$ by starting with $h'(x) = y$ and continuing to map all successors of x. Let $S(x, x') \in q_x^U$. If $g(x') \neq y'$, then extend h' to the subtree below x' by setting $h'(z) = g(z)$ for all $z \in \mathsf{var}(q_{x'})$. If $g(x') = y'$, then, by the induction hypothesis, there is a $p' \in \mathcal{F}_0(y')$ and a homomorphism h'' from $q_{x'}$ to $\mathcal{U}_{p',\mathcal{O}}$ with $h''(x') = y'$. Extend h' to the variables in $q_{x'}$ by mapping $q_{x'}$ according to h'' to the copy of p' that was attached to y in Point 3 of generalizing the subquery attached to $R(y, y')$.

  Now assume $\mathsf{func}(R) \in \mathcal{O}$. Then there is at most one $R(x, x') \in q_x^U$ with $g(x') = y'$. If there is none, choose an arbitrary $p \in \mathcal{F}_0(y)$ constructed by generalizing the subquery attached to $R(y, y')$ and extend h' as above. If there is a single such $R(x, x')$, then, by the induction hypothesis, there is a $p' \in \mathcal{F}_0(y')$ and homomorphism h'' from $q_{x'}^U$ to $\mathcal{U}_{p',\mathcal{O}}$ with $h''(x') = y'$. Let $p \in \mathcal{F}_0(y)$ be constructed by generalizing the subquery attached to $R(y, y')$ and attaching p' in Step 4, then extend h' as above.

- For every $R(y, y') \in q_y$, $q_{y'} \rightarrow (\mathcal{U}_{q_{x'}^U,\mathcal{O}}, x')$ for some $R(x, x') \in q_x^U$.

  Then there is an $A(y) \in q_y$ with $A(x) \notin \mathcal{U}_{q_x^U,\mathcal{O}}$ and we must even find an A with these properties and such that there is no $B(y) \in q_y$ with $\mathcal{O} \models B \sqsubseteq A$ and $\mathcal{O} \not\models A \sqsubseteq B$. Thus, Property (a) of dropping concept atoms is satisfied. To show that Property (b) is also satisfied, we have to argue that there is no $R(y, y') \in q_y$ with $\mathcal{O} \models \exists R \sqsubseteq A$. But by assumption for any such $R(y, y') \in q_y$ we have $q_{y'} \rightarrow (\mathcal{U}_{q_{x'}^U,\mathcal{O}}, x')$ for some $R(x, x') \in q_x^U$. This implies $A(x) \in \mathcal{U}_{q_x^U,\mathcal{O}}$, a contradiction.

  We may thus construct $p \in \mathcal{F}_0(y)$ by dropping the concept atom $A(y)$. Set $h'(x') = g(x')$ for all $x' \in \mathsf{var}(q_x^U)$.

This completes the proof of Claim 2.

By Claim 2, there is a $p' \in \mathcal{F}_0(x_0)$ such that $q^U \rightarrow \mathcal{U}_{p',\mathcal{O}}$. Let $p \in \mathcal{F}$ be the query that was obtained by applying the compensation step to p'. Then clearly also $q^U \to \mathcal{U}_{p,\mathcal{O}}$. Define h for all variables in U according to the homomorphism that witnesses this and let $\cdot^\downarrow$ be the extension of the original $\cdot^\downarrow$ for p to a homomorphism from $\mathcal{U}_{p,\mathcal{O}}$ to $\mathcal{U}_{q,\mathcal{O}}$.

We continue with the second step of the construction of h which covers parts of q' that are connected to the initial segment q^U and which are mapped to traces of $\mathcal{U}_{q,\mathcal{O}}$ rather than to $\mathsf{var}(q)$. Consider all atoms $R(x, x') \in q'$ with $h(x)$ defined, $h(x')$ undefined and $g(x') \notin \mathsf{var}(q)$. Before extending h to x', we first show that there is an atom $R(h(x), z) \in p$, added in Step 2A.

Since $g(x') \notin \mathsf{var}(q)$, $g(x')$ must be a trace $g(x)RM \in \Delta^{\mathcal{U}_{q,\mathcal{O}}}$ for some set M, hence $g(x) \leadsto^R_{q,\mathcal{O}} M$. This implies that there is no $R(g(x), y) \in q$ with $\mathcal{A}_q, \mathcal{O} \models \bigsqcap M(y)$. Furthermore, assume that there is a concept name B such that $\mathcal{O} \exists R \sqsubseteq B$ but $B(h(x)) \notin p$. Then, since q is $\mathcal{O}$-saturated, $B(g(x)) \in q$ and p must be the result of dropping the concept atom $B(g(x))$. However, since q' is $\mathcal{O}$-saturated, $B(x) \in q'$, contradicting Claim 2, therefore there is no such concept name.

Hence, Step 2A adds a fresh variable $z \in \mathsf{var}(p)$ and the atom $R(h(x), z)$ to p with $z^\downarrow = g(x)RM$. We extend h to the initial segment of $q'_{x'}$ that is mapped by g into the traces below $g(x')$. Set $h(\hat{x}) = zR_2M_2 \ldots R_nM_n$ for all $\hat{x}$ in this initial segment with $g(\hat{x}) = g(x)RMR_2M_2 \ldots R_nM_n$. If we reach a $R(x'', x''') \in q'_{x'}$ directed away from x_0 with $g(x'')$ a trace and $g(x''') \in \mathsf{var}(q)$, then leave $h(x''')$ undefined. The mapping h will be extended to the subtree $q'_{x'''}$ in the next steps. Note that $(*)$ is satisfied.

In the third step of the construction of h, we consider all $R(x, x') \in q'$ directed away from x_0 with $h(x)$ defined and $h(x')$ undefined. Before defining $h(x')$, we first show that

(a) there is an atom $R(h(x), y) \in p$ directed towards x_0 such that $\mathsf{func}(R^-) \notin \mathcal{O}$ and

(b) $g(x') \in \mathsf{var}(q)$.

Distinguish the following cases:

- $g(x)$ is a trace.

  Then $h(x)$ was defined in the second step. From the fact that $h(x')$ was not defined in the second step, it follows that $g(x') \in \mathsf{var}(q)$, as required for (b). Consequently, $g(x)$ must be of the form $g(x')R^-M \in \mathcal{U}_{q,\mathcal{O}}$ and $h(x)$ was defined in the second step to be a fresh variable added to p in Step 2A of its construction and this variable is an R^--successor of some variable y, that is, $R^-(y, h(x)) \in p$ directed away from x_0. We may thus use the inverse of this atom as the desired atom $R(h(x), y)$ in (a). Since $g(x')R^-M$ is a trace in $\mathcal{U}_{q,\mathcal{O}}$ and due to the syntactic restriction adopted by the *DL-Lite*$^{\mathcal{F}-}$ ontology $\mathcal{O}$, we further have $\mathsf{func}(R) \notin \mathcal{O}$.

- $g(x)$ is not a trace, that is, $g(x) \in \mathsf{var}(q)$.

  Since $h(x')$ has neither been defined in the first nor in the second step, we must have $x \in U$, $g(x') \in \mathsf{var}(q)$ (as required for (b)) and $R(g(x), g(x')) \in q$ is directed towards x_0. The latter implies that x is not the root of q', thus q' contains an atom $S(x'', x)$ directed away from x_0. From $x \in U$, it follows by definition of U that $x'' \in U$. Thus $h(x'')$ and $h(x')$ were both defined in the first step and, due to the formulation of that step, $S(x'', x) \in q'$ directed away from x_0 implies that q contains the atom $S(g(x''), g(x))$ directed away from x_0. So $S(g(x''), g(x)) \in q$ is directed away from x_0 and $R(g(x), g(x')) \in q$ is directed towards x_0. Since q is a tree, this implies $g(x'') = g(x')$ and $S = R^-$. We have thus shown that $R^-(x'', x)$ and $R(x, x')$ are atoms in q' that are both directed away from x_0. Since q' satisfies all functionality assertions in $\mathcal{O}$, this implies $\mathsf{func}(R) \notin \mathcal{O}$. We use $R(h(x), h(x'')) \in p$ as the desired atom $R(h(x), y)$.

Consider the inverse $R^-(y, h(x))$ of the atom $R(h(x), y) \in p$ that exists due to (a). In the start of Step 2B of the construction of p, the inverse atom $R^-(y, h(x))$ is considered and leads to the introduction of an atom $R(h(x), y')$, y' a fresh variable with $y'^\downarrow = y^\downarrow$. Set $h(x') = y'$. Note that $R(h(x), h(x')) \in p$ was marked in Step 2B of the construction of p.

In the final step of the construction of h we define $h(x)$ for all remaining variables x. We do this by repeatedly choosing atoms $R(x, x') \in q'$ directed away from x_0 such that

1. $h(x)$ and $h(x')$ defined and
2. for all $S(x', x'') \in q'$ directed away from x_0, $h(x'')$ is undefined and there is at least one such $S(x', x'')$.

If we choose such an $R(x, x')$ directly after the third step of the construction of h, then $g(x') \in \mathsf{var}(q)$ due to (b) and $R(h(x), h(x')) \in p$ was marked in Step 2B of the construction of p. We implement our extension of h such that these conditions are always guaranteed when we choose an $R(x, x') \in q'$ that satisfies Properties 1 and 2 above.

Let $R(x, x') \in q'$ be an atom that satisfies Properties 1 and 2. First assume that $\mathsf{func}(R^-) \notin \mathcal{O}$. Then processing the marked atom $R(h(x), h(x')) \in p$ in Step 2B of the construction of p results in a copy $\widehat{q}$ of q to be added to p, with the copy of $h(x')^\downarrow$ in $\widehat{q}$ glued to $h(x')$. Define h for all variables $x'' \in \mathsf{var}(q'_{x'})$ by setting $h(x'')$ to be the copy of $g(x'')$ in $\widehat{q}$ if $g(x'')$ is a variable, or to be the trace $h(x''')R_1M_1 \ldots R_nM_n$ if $g(x'')$ is the trace $g(x''')R_1M_1 \ldots R_nM_n$.

Now assume that $\mathsf{func}(R^-) \in \mathcal{O}$. Consider each $S(x', x'') \in q'$ directed away from x_0. We distinguish two cases:

- $g(x'')$ is a trace.

  Since $g(x') \in \mathsf{var}(q)$, $g(x'')$ must be of the form $g(x')SM$. Thus, $g(x') \leadsto^S_{q,\mathcal{O}} M$ and by $(*)$, $h(x')^\downarrow \leadsto^S_{q,\mathcal{O}} M$ and when the marked atom $R(h(x), h(x')) \in p$ is processed in Step 2B of the construction of p. Thus, Point (iii) of Step 2B adds to p atoms $S(h(x'), u)$, $S^-(u, y')$, and $A(u)$ for every $A \in M$. Additionally, processing the marked atom $S^-(u, y')$ attaches a copy $\widehat{q}$ of q to y', since $y'^\downarrow = g(x') \leadsto^S_{q,\mathcal{O}} M$ and therefore either $\mathsf{func}(S) \notin \mathcal{O}$ or there is no atom $S(y'^\downarrow, z) \in q$.

  Extend h by setting $h(x''') = uR_2M_2 \ldots R_nM_n$ if $g(x''') = g(x')SMR_2M_2 \ldots R_nM_n$ for all $x''' \in \mathsf{var}(q'_{x''})$ up until $g(x''') = g(x')$. If there is a subtree

$q'_{x'''}$ with $x''' \in \mathsf{var}(q'_{x''})$ and $g(x''') = g(x')$, map it to y' and the attached $\widehat{q}$ by setting $h(x''') = y'$ and all $h(z)$ for $z \in \mathsf{var}(q'_{x'''})$ to the copy of $g(z)$ in $\widehat{q}$, or the trace starting in $\widehat{q}$.

- $g(x'')$ is not a trace.

  Since $g(x') \in \mathsf{var}(q)$, this implies $S(g(x'), g(x'')) \in q$. It follows from $(*)$ that $h(x')^{\downarrow} = g(x')$ and thus $S(h(x')^{\downarrow}, g(x'')) \in q$. When the marked atom marked atom $R(h(x), h(x')) \in p$ is processed in Step 2B of the construction of p, then $S(h(x')^{\downarrow}, g(x'')) \in q$ is thus one of the atoms under consideration in Point (ii).

  Since $R(x, x') \in q'$ and $S(x', x'') \in q'$ are both directed away from x_0 and q' satisfies all functionality assertions in $\mathcal{O}$, we have $S \neq R$. It follows from $(*)$ that $h(x)^{\downarrow} = g(x)$ As a consequence $S(h(x')^{\downarrow}, g(x'')) \neq R^{-}(h(x')^{\downarrow}, h(x)^{\downarrow})$. Thus, in Point (ii) of Step 2B an atom $S(h(x'), z')$ is added to p with z' a fresh variable and $z'^{\downarrow} = g(x'')$ and this atom is marked. Set $h(x'') = z'$ and leave the successors of x'' in q' to be processed in subsequent iterations of the loop in step four of the construction of h.

This completes the construction of h and the proof of Condition 3. ❏

Lemma 6. *The construction of $\mathcal{F}$ runs in time polynomial in $||q|| + ||\mathcal{O}||$ (and thus $\sum_{p \in \mathcal{F}} ||p||$ is polynomial in $||q|| + ||\mathcal{O}||$).*

Proof. In order to reduce notational clutter, we introduce some abbreviations used throughout the proof.

- $s = |\mathsf{sig}(q)|$ denotes the number of concept and role names used in q;
- $o = ||\mathcal{O}||$ denotes the size of $\mathcal{O}$;
- for an ELIQ p, $n_p = |\mathsf{var}(p)|$ denotes the number of variables in p;
- for a set Q of queries, n_Q denotes $\sum_{p \in Q} n_p$.

We assume without loss of generality that s and o are at least one.

We start with analyzing the size of the queries in $\mathcal{F}_0(x)$ that are obtained as the result of the 'generalize' step.

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

1. $|\mathcal{F}_0(x)| \leq s \cdot n_{q_x}$;
2. $n_{\mathcal{F}_0(x)} \leq s \cdot n_{q_x}^3$.

Proof of the claim. The proof of both points is by induction on the codepth of x in q. We start with Point 1. For the base case, consider a variable x of codepth 0 in q, that is, a leaf. In this case, only Step (A) is applicable, and it adds at most s queries to $\mathcal{F}_0(x)$.

For the inductive step, consider a variable x of codepth greater than 0. We partition $\mathcal{F}_0(x)$ into $\mathcal{F}_0^A(x)$ and $\mathcal{F}_0^B(x)$, that is, the queries that are obtained by dropping a concept atom in Step (A) and the queries that are obtained by generalizing a subquery in Step (B), respectively, and analyze them separately, starting with $\mathcal{F}_0^A(x)$. Clearly, there are at most s queries in $\mathcal{F}_0^A$, that is,

$$|\mathcal{F}_0^A(x)| \leq s.$$

Next, we analyze $\mathcal{F}_0^B(x)$. Each query in $\mathcal{F}_0^B(x)$ is obtained by first picking, in Point 1, an atom $R(x, y)$ in q_x. If $\mathsf{func(R)} \notin \mathcal{O}$, we add 1 query to $\mathcal{F}_0^B(x)$. Otherwise, we add $|\mathcal{F}_0(y)|$ queries (in Point 4). Thus, we obtain

$$|\mathcal{F}_0^B(x)| \leq |\{R(x,y) \in q_x \mid \mathsf{func}(R) \notin \mathcal{O}\}| + \sum_{\substack{R(x,y) \in q_x, \\ \mathsf{func}(R) \in \mathcal{O}}} |\mathcal{F}_0(y)|$$

Using the fact that $n_{q_y} \geq 1$ and the induction hypothesis, we obtain

$$|\mathcal{F}_0^B(x)| \leq \sum_{R(x,y) \in q_x} s \cdot n_{q_y} = s \cdot \sum_{R(x,y) \in q_x} n_{q_y}.$$

The above sum can be simplified to $n_{q_x} - 1$. Hence, we obtain

$$|\mathcal{F}_0(x)| = |\mathcal{F}_0^A(x)| + |\mathcal{F}_0^B(x)| \leq s + s \cdot (n_{q_x} - 1) = s \cdot n_{q_x}.$$

We now prove Point 2, again by induction on the codepth of x in q. For the base case, consider a variable x of codepth 0 in q, that is, a leaf. In this case, only Step (A) is applicable, and it adds at most s queries of size 1 to $\mathcal{F}_0(x)$.

For the inductive step, consider a variable x of codepth greater than 0 and the same partition of $\mathcal{F}_0(x)$ into $\mathcal{F}_0^A(x)$ and $\mathcal{F}_0^B(x)$ as before. Clearly, every $p \in \mathcal{F}_0^A(x)$ uses n_{q_x} variables and there are at most s queries in $\mathcal{F}_0^A$. Thus, we have

$$n_{\mathcal{F}_0^A(x)} \leq s \cdot n_{q_x}.$$

Next, we analyze $\mathcal{F}_0^B(x)$. Each query in $\mathcal{F}_0^B(x)$ is obtained by first picking, in Point 1, an atom $R(x, y)$ in q_x. If $\mathsf{func(R)} \notin \mathcal{O}$, we add $\sum_{p \in \mathcal{F}_0(y)} n_p$ variables (in Point 3). Otherwise, we replace q_y with some element of $\mathcal{F}_0(y)$ (in Point 4). Thus, we obtain

$$\begin{aligned} n_{\mathcal{F}_0^B(x)} \leq & \sum_{\substack{R(x,y) \in q_x, \\ \mathsf{func}(R) \notin \mathcal{O}}} (n_{q_x} + n_{\mathcal{F}_0(y)}) + \\ & \sum_{\substack{R(x,y) \in q_x, \\ \mathsf{func}(R) \in \mathcal{O}}} (n_{q_x} \cdot |\mathcal{F}_0(y)| + n_{\mathcal{F}_0(y)}) \\ \leq & \sum_{R(x,y) \in q_x} (n_{q_x} \cdot |\mathcal{F}_0(y)| + n_{\mathcal{F}_0(y)}). \end{aligned}$$

Plugging in the induction hypothesis from both Point 1 and 2, we obtain

$$\begin{aligned} n_{\mathcal{F}_0^B(x)} &\leq \sum_{R(x,y) \in q_x} (n_{q_x} \cdot s \cdot n_{q_y} + s \cdot n_{q_y}^3) \\ &= s \cdot n_{q_x} \sum_{R(x,y) \in q_x} n_{q_y} + s \sum_{R(x,y) \in q_x} n_{q_y}^3. \qquad (2) \end{aligned}$$

We simplify the right-hand side of (2) by making the following observations:

- $\sum_{R(x,y)\in q_x} n_{q_y} = n_{q_x} - 1$, and
- $\sum_{R(x,y)\in q_x} n_{q_y}^3 \leq \left(\sum_{R(x,y)\in q_x} n_{q_y}\right)^3 = (n_{q_x} - 1)^3$. Here, the inequality is an application of the general inequality $\sum_i a_i^3 \leq (\sum_i a_i)^3$, for every sequence of non-negative numbers $a_1, \dots, a_k$.

Using these observations, Inequality (2) can be simplified to:

$$\begin{aligned} n_{\mathcal{F}_0^B(x)} &\leq s \cdot n_{q_x} \cdot (n_{q_x} - 1) + s \cdot (n_{q_x} - 1)^3 \\ &= s \cdot \left(n_{q_x}^3 - 2n_{q_x}^2 + 2n_{q_x} - 1\right). \end{aligned}$$

Overall, we get

$$\begin{aligned} n_{\mathcal{F}_0(x)} &= n_{\mathcal{F}_0^A(x)} + n_{\mathcal{F}_0^B(x)} \\ &\leq s \cdot n_{q_x} + s \cdot \left(n_{q_x}^3 - 2n_{q_x}^2 + 2n_{q_x} - 1\right) \\ &= s \cdot \left(n_{q_x}^3 - 2n_{q_x}^2 + 3n_{q_x} - 1\right) \\ &\leq s \cdot n_{q_x}^3. \end{aligned}$$

In the last inequality, we used that $z^3 \geq z^3 - 2z^2 + 3z - 1$, for all numbers $z \geq 1$. This finishes the proof of the claim.

We analyze now the compensation Step 2, in which the queries in $\mathcal{F}_0(x_0)$ are further extended. We let $\mathcal{F}_1$ denote the result of applying Step 2A to $\mathcal{F}_0(x_0)$. In Step 2A, we add at most one variable per variable in $\mathcal{F}_0(x_0)$ and concept $\exists R.B$ that occurs in $\mathcal{O}$. Therefore, we add at most $n_{\mathcal{F}_0(x_0)} \cdot o$ variables. Using the claim, we get

$$n_{\mathcal{F}_1} \leq n_{\mathcal{F}_0(x_0)} + n_{\mathcal{F}_0(x_0)} \cdot o \leq s \cdot n_q^3 \cdot (1 + o).$$

We now analyze Step 2B, applied to some query $p \in \mathcal{F}_1$. First of all note that the marking proviso "if $R(x,y)$ is marked then $y^\downarrow$ is defined and if $x^\downarrow$ is undefined, then $\mathsf{func}(R^-) \notin \mathcal{O}$ or q contains no atom of the form $R(y^\downarrow, z)$" is indeed satisfied.

Consider now an atom $R(x,y) \in p$ that was marked in the *Start* phase. We distinguish two cases.

- If $x^\downarrow$ is undefined, then the marking proviso implies that $\mathsf{func}(R^-) \notin \mathcal{O}$ or q contains no atom of the form $R(y^\downarrow, z)$. In the *Step* phase, we just unmark the atom and add a copy of q, hence no iteration takes place, and the query size increases by n_q.
- Otherwise, $x^\downarrow$ is defined, and by definition of $^\downarrow$, we have $R(x^\downarrow, y^\downarrow) \in q$. Now, the iterative process ensures that:
  - Whenever an atom $S(y, z')$ is marked in (ii), then both $y^\downarrow$ and $z'^\downarrow$ are defined and $S(y^\downarrow, z'^\downarrow) \in q$. Moreover, the condition '$S(y^\downarrow, z) \neq R^-(y^\downarrow, x^\downarrow)$' and the fact that q is an ELIQ ensure that every atom from q is 'met' at most once during the entire process.
  - Whenever an atom $S^-(u, y')$ is marked in (iii), then $u^\downarrow$ is undefined. Hence, the marking proviso implies that, in the *Step* phase, this atom is unmarked, a copy of q is added, and the iteration stops.

Overall, we obtain that, per role atom in p, the marking process adds at most n_q role atoms in Step (ii), for each such atom and every $\exists r.B$ in $\mathcal{O}$ one more role atom in Step (iii), and for each introduced variable at most one copy of q. All this is polynomial in $||q||$ and $||\mathcal{O}||$. Moreover, the computation of $\mathcal{F}$ can be carried out in polynomial time since all the involved queries are of polynomial size and consequences of $\mathcal{O}$ can be decided in polynomial time. ❑

E Proofs for Section 5

Theorem 5. *Let $\mathcal{O}$ be an ontology formulated in DL-Lite$^{\mathcal{H}}$ or DL-Lite$^{\mathcal{F}-}$. Then for every ELIQ q that is satisfiable w.r.t. $\mathcal{O}$, there are sets of data examples (E^+, E^-) that uniquely characterize q w.r.t. $\mathcal{O}$ and such that $||(E^+, E^-)||$ is polynomial in $||q|| + ||\mathcal{O}||$. If $\mathcal{O}$ is a DL-Lite$^{\mathcal{F}-}$ ontology, then (E^+, E^-) can be computed in polynomial time and the same holds for DL-Lite$^{\mathcal{H}}$ if q is $\mathcal{O}$-minimal.*

Proof. Let $\mathcal{O}$ and $q(x)$ be as in the theorem. By Theorems 1 and 4, 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 $E^- = \{(\mathcal{A}_p, x) \mid p \in \mathcal{F}_q(x)\}$. It is not hard to verify that q fits (E^+, E^-). We show that (E^+, E^-) in fact uniquely characterizes q w.r.t. $\mathcal{O}$.

Let q' be an ELIQ that fits (E^+, E^-). 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 $q' \subseteq_{\mathcal{O}} q$. Thus $q' \equiv_{\mathcal{O}} q$, as required. ❑

F Proofs for Section 6

We start with showing how to construct a seed CQ in the case that the ontology $\mathcal{O}$ contains no concept disjointness constraints. This is in fact trivial if $\mathcal{O}$ contains no role disjointness constraint either, as then we can simply use

$$q_H^0(x_0) = \bigwedge_{A \in \Sigma \cap \mathsf{N_C}} A(x_0) \wedge \bigwedge_{r \in \Sigma \cap \mathsf{N_R}} r(x_0, x_0).$$

Here and in what follows, for brevity we use Σ to denote $\mathsf{sig}(\mathcal{O})$.

We consider now the case with role disjointness constraints (but still without concept disjointness). Let $\mathbf{R} = \{r_1, \dots, r_m\}$ be the set of all role names $r \in \Sigma$ such that $\exists r$ is satisfiable w.r.t. $\mathcal{O}$. If, for example, $\mathcal{O}$ contains $r \sqsubseteq s$ and $r \sqcap s \sqsubseteq \bot$, then $\exists r$ is not satisfiable w.r.t. $\mathcal{O}$.

Introduce variables $x_0, \dots, x_{2m}$ and let K_{2m+1} be the $2m + 1$-clique that uses these variables as its vertices. It is known that for each $n \geq 1$, the n-clique K_n has at least $\frac{n-1}{2}$ Hamilton cycles that are pairwise edge-disjoint, see for instance the survey [Alspach *et al.*, 1987]. We thus find in K_{2m+1} Hamilton cycles $P_1, \dots, P_m$ that are pairwise edge-disjoint. By directing the cycles, we may view each P_i as a set of directed edges (x_j, x_ℓ). We then set

$$\begin{aligned} q_H^0(x_0) \quad = \quad & \bigwedge_{A \in \Sigma \cap \mathsf{N_C},\ 0 \leq i \leq 2m} A(x_i) \\ & \bigwedge_{(x_i,x_j) \in P_1} r_1(x_i, x_j) \wedge \cdots \wedge \bigwedge_{(x_i,x_j) \in P_m} r_m(x_i, x_j). \end{aligned}$$

Clearly, q_H^0 has no multi-edges and thus satisfies all role disjointness constraints in $\mathcal{O}$. Moreover, every variable has exactly one r-successor and exactly one r-predecessor for every role name $r \in \mathbf{R}$. On the one hand, this implies that all functionality assertions in $\mathcal{O}$ are satisfied. On the other hand, it means that there is a homomorphism from every target ELIQ q_T to q_H^0 because any q_T is required to be satisfiable w.r.t. $\mathcal{O}$ and thus may only use role names from $\mathbf{R}$.

If $\mathcal{O}$ contains at least one concept disjointness constraint $B_1 \sqcap B_2 \sqsubseteq \bot$, then we cannot use the above q_H^0 as it is not satisfiable w.r.t. $\mathcal{O}$, but we may obtain a seed query q_H^0 by viewing $B_1 \sqcap B_2$ as an ELIQ q in the obvious way and posing q as an equivalence query to the oracle. Since the target query is satisfiable w.r.t. $\mathcal{O}$, the oracle is forced to return a positive counterexample $(\mathcal{A}, a)$, that is, a pair $(\mathcal{A}, a)$ such that $\mathcal{A}, \mathcal{O} \models q_T(a)$ and $\mathcal{A}, \mathcal{O} \not\models q_H(a)$. The desired query q_H^0 is $(\mathcal{A}, a)$ viewed as a CQ with answer variable a. Note that when learning with equivalence queries, then in polynomial time learnability, the running time of the learning algorithm may also polynomially depend, at any given time, on the size of the largest counterexample returned by the oracle so far. This condition is satisfied by our algorithm.

Lemma 7. *In DL-Lite$^\mathcal{H}$ and DL-Lite$^{\mathcal{F}-}$, every polynomial time learning algorithm for ELIQs under ontologies in normal form that uses only membership queries can be transformed into a learning algorithm with the same properties for ELIQs under unrestricted ontologies.*

Proof. We show the lemma by converting a learning algorithm L' for ontologies in normal form into a learning algorithm L for unrestricted ontologies, relying on the normal form described in Lemma 12. Since L will ask a single query for every query asked by L', the lemma follows.

We start with *DL-Lite*$^\mathcal{H}$. Given a *DL-Lite*$^\mathcal{H}$ ontology $\mathcal{O}$ and a signature $\Sigma = \mathsf{sig}(\mathcal{O})$ with $\mathsf{sig}(q_T) \subseteq \Sigma$, algorithm L first computes the ontology $\mathcal{O}'$ in normal form as per Lemma 12, 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 $\mathcal{A}', \mathcal{O}' \models q_T(a)$, we may assume that $\mathcal{A}'$ satisfies the functionality assertions from $\mathcal{O}$, since otherwise the answer is trivially "yes". Then, L instead asks the membership query $\mathcal{A}, \mathcal{O} \models q_T(a)$, where $\mathcal{A}$ is obtained from $\mathcal{A}'$ as follows. Start with $\mathcal{A} = \mathcal{A}'$, and

$(*)$ add $C(b)$, for each concept assertion $X_C(b) \in \mathcal{A}'$.

Here, the addition of $C(b)$ for an $\mathcal{ELI}$-concept C to an ABox $\mathcal{B}$ is defined as expected in case of *DL-Lite*$^\mathcal{H}$ ontologies: View $C(b)$ as a tree-shaped ABox $\mathcal{A}_{C(b)}$ with root b and assume without loss of generality that b is the only individual shared by $\mathcal{B}$ and $\mathcal{A}_{C(b)}$. Then take the union of $\mathcal{A}$ and $\mathcal{A}_{C(b)}$.

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. For "if", suppose that $\mathcal{A}, \mathcal{O} \models q(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}'$ and $\mathcal{O}'$. We can assume that $\Delta^\mathcal{I}$ does not mention any of the individuals that were introduced in the construction of $\mathcal{A}$. We will construct a model $\mathcal{I}'$ of $\mathcal{A}$ and $\mathcal{O}$ that has a homomorphism h from $\mathcal{I}'$ to $\mathcal{I}$ which is the identity on $\Delta^\mathcal{I}$. This clearly suffices since $\mathcal{I}' \models q(a)$.

The interpretation $\mathcal{I}'$ has the following domain:

$$\Delta^{\mathcal{I}'} = \Delta^\mathcal{I} \cup \bigcup_{X_C(b) \in \mathcal{A}'} \mathsf{ind}(\mathcal{A}_{C(b)})$$

In order to define the interpretation of concept and role names, observe first that, for every $X_C(b) \in \mathcal{A}'$, there is a homomorphism $h_{C(b)} : \mathcal{A}_{C(b)}, b \to \mathcal{I}, b$ since $\mathcal{I}$ is a model of $\mathcal{A}'$ and $\mathcal{O}'$, and $\mathcal{O}' \models X_C \sqsubseteq C$. We combine all these homomorphisms into a mapping $h : \Delta^{\mathcal{I}'} \to \Delta^\mathcal{I}$ by taking

$$h(c) = \begin{cases} c & \text{if } c \in \Delta^\mathcal{I}, \\ h_{C(b)}(c) & \text{if } c \in \mathsf{ind}(\mathcal{A}_{C(b)}) \setminus \Delta^\mathcal{I}. \end{cases}$$

Then, we set

$$A^{\mathcal{I}'} = \{d \mid h(d) \in A^\mathcal{I}\}$$
$$r^{\mathcal{I}'} = \{(d, e) \mid (h(d), h(e) \in r^\mathcal{I}\}$$

It is routine to verify that $\mathcal{I}'$ is as required.

For "only if", suppose that $\mathcal{A}', \mathcal{O}' \models q(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}$ and $\mathcal{O}$. Observe that the model $\mathcal{I}'$ of $\mathcal{O}'$ that can be obtained from $\mathcal{I}$ as in Lemma 1 Point 2 coincides with $\mathcal{I}$ on Σ and is additionally a model of $\mathcal{A}'$. It follows that $\mathcal{I} \models q(a)$ as required. This finishes the proof of Claim 1.

In the case of *DL-Lite*$^{\mathcal{F}-}$ ontologies, we follow the same strategy. However, the addition of $\mathcal{A}_{C(b)}$ in $(*)$ has to respect the functionality assertions. In fact, not even $\mathcal{A}_{C(b)}$ necessarily satisfies the functionality assertions in $\mathcal{O}$. We define the *addition of a tree-shaped ABox $\mathcal{B}$ with root b_0 to $\mathcal{A}$ at a* inductively on the structure of $\mathcal{B}$ as follows:

1. for all $A(b_0) \in \mathcal{B}$, add $A(a)$ to $\mathcal{A}$;
2. for all $R(b_0, b') \in \mathcal{B}$, let $\mathcal{B}'$ be the sub-ABox of $\mathcal{B}$ rooted at b' and
   (a) if $\mathsf{func}(R) \in \mathcal{O}$ and there is an atom $R(a, a')$ in $\mathcal{A}$, then add $\mathcal{B}'$ to $\mathcal{A}$ at a';
   (b) otherwise, add an atom $R(a, a')$ for a fresh individual a' and add $\mathcal{B}'$ to $\mathcal{A}$ at a'.

Note that there can only be one atom $R(a, a')$ in Step 2(a) if $\mathcal{A}$ satisfies the functionality assertions in $\mathcal{O}$. It should be also clear that the resulting ABox also satisfies the functionality assertions if $\mathcal{A}$ does.

Now, $\mathcal{A}$ is obtained from $\mathcal{A}'$ by starting with $\mathcal{A} = \mathcal{A}'$ and

$(*')$ adding $\mathcal{A}_{C(b)}$ to $\mathcal{A}$ at b, for each $X_C(b) \in \mathcal{A}'$.

By the following claim, the answer to the modified membership query coincides with that to the original query.

Claim 2. $\mathcal{A}', \mathcal{O}' \models q(a)$ iff $\mathcal{A}, \mathcal{O} \models q(a)$ for all ELIQs q that only use symbols from Σ.

Proof of Claim 2. For "if", suppose that $\mathcal{A}, \mathcal{O} \models q(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}'$ and $\mathcal{O}'$. We can assume that $\Delta^\mathcal{I}$ does not mention any of the individuals that were introduced in the construction of $\mathcal{A}$. We will construct a model $\mathcal{I}'$ of $\mathcal{A}$ and $\mathcal{O}$ that has a homomorphism h from $\mathcal{I}'$ to $\mathcal{I}$ which is the identity on $\Delta^{\mathcal{I}}$. This clearly suffices since $\mathcal{I}' \models q(a)$.

Observe first that, for every $X_C(b) \in \mathcal{A}'$, there is a homomorphism $h_{C(b)} : \mathcal{A}_{C(b)}, b \rightarrow \mathcal{I}, b$ since $\mathcal{I}$ is a model of $\mathcal{A}'$ and $\mathcal{O}'$, and $\mathcal{O}' \models X_C \sqsubseteq C$.

Let F denote the set of fresh individuals introduced in the construction of $\mathcal{A}$. Note that for every fresh element there is an $X_C(b) \in \mathcal{A}'$ which 'triggered' the addition of d in some (possibly later) application of Step 2(b). We associate with every $d \in F$ an element $g(d) \in \Delta^{\mathcal{I}}$ as follows:

- if d was introduced in Step 2(b) triggered by $X_C(b) \in \mathcal{A}'$, then set $g(d) = h_{C(b)}(b')$ where b' is the element mentioned in Step 2.

We further associate with every $d \in F$ a tree-shaped interpretation $\mathcal{I}_d$. Intuitively, $\mathcal{I}_d$ is the unraveling of $\mathcal{I}$ at $g(d)$, with the functionality assertions taken into account. Formally, the domain $\Delta^{\mathcal{I}_d}$ consists of all sequences $a_0 R_1 a_1 \dots R_n a_n$ such that

- $a_0 = g(d)$;
- $a_i \in \Delta^{\mathcal{I}}$, for all i with $0 \leq i \leq n$;
- $(a_i, a_{i+1}) \in R_{i+1}^{\mathcal{I}}$, for all i with $0 \leq i < n$;
- if $\mathsf{func}(R_i^-) \in \mathcal{O}$, then $R_{i+1} \neq R_i^-$, for all i with $0 \leq i < n$;
- if $\mathsf{func}(R_1) \in \mathcal{O}$, there is no atom of shape $R_1(d, d')$ in $\mathcal{A}_{C(b)}$, where $X_C(b) \in \mathcal{A}'$ triggered the addition of d.

The interpretation of concept and role names is as follows:

$$A^{\mathcal{I}_d} = \{a_0 R_1 a_1 \dots R_n a_n \in \Delta^{\mathcal{I}_d} \mid a_n \in A^{\mathcal{I}}\} \quad \text{for all } A \in \mathsf{N_C};$$
$$r^{\mathcal{I}_d} = \{(\pi, \pi r a) \mid \pi r a \in \Delta^{\mathcal{I}_d}\} \cup \{(\pi r^- a, \pi) \mid \pi r^- a \in \Delta^{\mathcal{I}_d}\} \quad \text{for all } r \in \mathsf{N_R}.$$

The interpretation $\mathcal{I}'$ is then obtained by starting with $\mathcal{I}' = \mathcal{I} \cup \mathcal{A}$, and then adding, for every $d \in F$, a copy $\widehat{\mathcal{I}_d}$ of $\mathcal{I}_d$ and gluing the copy of $g(d)$ in $\widehat{\mathcal{I}_d}$ to d.

It is routine to verify that $\mathcal{I}'$ is as required.

For "only if", suppose that $\mathcal{A}', \mathcal{O}' \models q(a)$ and let $\mathcal{I}$ be a model of $\mathcal{A}$ and $\mathcal{O}$. Observe that the model $\mathcal{I}'$ of $\mathcal{O}'$ that can be obtained from $\mathcal{I}$ as in Lemma 1 Point 2 coincides with $\mathcal{I}$ on Σ and is additionally a model of $\mathcal{A}'$. It follows that $\mathcal{I} \models q(a)$ as required. This finishes the proof of Claim 2. ❑

We now work towards showing that the algorithm presented in Section 6 indeed learns ELIQs under ontologies formulated in *DL-Lite*$^{\mathcal{H}}$ or *DL-Lite*$^{\mathcal{F}-}$, in polynomial time. We start with analyzing the minimize subroutine.

Lemma 13. *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 $q_T(x_0)$, and let $q'(x_0) = \mathsf{minimize}(q)$. Then*

1. *$q \subseteq_{\mathcal{O}} q'$ and $q' \subseteq_{\mathcal{O}} q_T$;*
2. *$|\mathsf{var}(q')| \leq |\mathsf{var}(q_T)|$;*
3. *q' is $\mathcal{O}$-minimal, connected, and $\mathcal{O}$-saturated.*

Proof. We start with Point 1. We have $q \subseteq_{\mathcal{O}} q'$ since q' is a subset of q. For $q' \subseteq_{\mathcal{O}} q_T$, it suffices to observe that minimize ensures in each step that $\mathcal{A}_{q'}, \mathcal{O} \models q_T(x_0)$.

For Point 2, it suffices to show that $\mathsf{var}(q') \subseteq \mathsf{img}(h)$ for every homomorphism h from q_T to $\mathcal{U}_{q',\mathcal{O}}$ with $h(x_0) = x$. Assume for a contradiction that there is a homomorphism h from q_T to $\mathcal{U}_{q',\mathcal{O}}$ with $h(x_0) = x_0$ and a $y \in \mathsf{var}(q')$ that is not in $\mathsf{img}(h)$. Choose some $r(x, y) \in q$ such that the distance from x_0 to x is strictly smaller than that from x_0 to y, and let q^- be the restriction of $q \setminus \{r(x, y)\}$ to the atoms that contain only variables reachable from x_0 in $q \setminus \{r(x, y)\}$. We argue that h is a homomorphism from q_T to $\mathcal{U}_{q^-,\mathcal{O}}$ which witnesses that $\mathcal{A}_{q^-}, \mathcal{O} \models q_T(x_0)$, in contradiction to the construction of q' and $y \in \mathsf{var}(q')$.

To see that h is a homomorphism, first note that since q_T is connected and $h(x_0) = x_0$, the range of h contains only variables from $\mathsf{var}(q^-)$ and the subtrees below them that (consist of traces and) are added in the construction of the universal model. Next observe that for all $x_1, x_2 \in \mathsf{var}(q^-)$, the following holds by construction of universal models and since q is $\mathcal{O}$-saturated:

(a) $A(x_1) \in \mathcal{U}_{q',\mathcal{O}}$ iff $A(x_1) \in \mathcal{U}_{q^-,\mathcal{O}}$;

(b) $r(x_1, x_2) \in \mathcal{U}_{q',\mathcal{O}}$ iff $r(x_1, x_2) \in \mathcal{U}_{q^-,\mathcal{O}}$.

Since $\mathcal{O}$ is in normal form and due to (a), it follows from the construction of universal models that the subtree in $\mathcal{U}_{q',\mathcal{O}}$ below each $x_1 \in \mathsf{var}(q^-) \setminus \{y\}$ is identical to the subtree in $\mathcal{U}_{q^-,\mathcal{O}}$ below x_1. Moreover, the subtree in $\mathcal{U}_{q',\mathcal{O}}$ below y can be obtained from the subtree in $\mathcal{U}_{q^-,\mathcal{O}}$ below y by dropping subtrees. It should thus be clear that, as required, h is a homomorphism from q_T to $\mathcal{U}_{q^-,\mathcal{O}}$.

For Point 3, we start with $\mathcal{O}$-minimality. Assume for a contradiction that q' is not $\mathcal{O}$-minimal, that is, there is a homomorphism h from q' to $\mathcal{U}_{q'',\mathcal{O}}$ with $h(x_0) = x_0$ where $q'' = q|_{\mathsf{var}(q)\setminus\{y\}}$ for some variable $y \in \mathsf{var}(q)$. Choose some $r(x, y) \in q$ such that the distance from x_0 to x is strictly smaller than that from x_0 to y and let q^- be the restriction of $q \setminus \{r(x, y)\}$ to the atoms that contain only variables reachable from x_0 in $q \setminus \{r(x, y)\}$. We can show as above that h is a homomorphism from q_T to $\mathcal{U}_{q^-,\mathcal{O}}$ and thus $\mathcal{A}_{q^-}, \mathcal{O} \models q_T(x_0)$, in contradiction to the construction of q' and $y \in \mathsf{var}(q')$.

Now for connectedness. Assume for a contradiction that $q'(x_0)$ is not connected and let x be a variable that is in a different maximally connected component of q' than x_0. Then x is also in a different maximally connected component of $\mathcal{U}_{q',\mathcal{O}}$ than x_0. By Point 1, there is a homomorphism h from q_T to $\mathcal{U}_{q',\mathcal{O}}$ with $h(x_0) = x_0$. Since q_T is connected, we must have $x \notin \mathsf{img}(h)$, thus $\mathsf{var}(q') \not\subseteq \mathsf{img}(h)$. But we have already seen in the proof of Point 2 that this is impossible.

Finally, $\mathcal{O}$-saturatedness of q' is clear given that the original CQ q is $\mathcal{O}$-saturated, q' is a subquery of q, and during the construction of q' we have not removed any concept atoms on any of the remaining variables. ❑

We next turn to the treeify subroutine. We start with a preliminary.

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 lemma gives an important property of simulations. The proof is standard and omitted.

Lemma 14. *Let $\mathcal{O}$ be a DL-Lite$^{\mathcal{HF}}$ 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 15. *Let $q(x_0)$ be a unary CQ that is $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$ such that $q \subseteq_{\mathcal{O}} q_T$ for the target query $q_T(x_0)$. Further let $p_1(x_0), p_2(x_0), \ldots$ be the sequence of CQs computed by* $\mathsf{treeify}(q)$. *Then for all $i \geq 1$,*

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

Proof. We show Point 1 by induction on i. The case $i = 1$ is immediate by Point 1 of Lemma 13 since $p_1 = \mathsf{minimize}(q)$. Now let $i \geq 1$. By the induction hypothesis $p_i \subseteq q_T$ and thus $\mathcal{A}_{p_i}, \mathcal{O} \models q_T(x_0)$. By construction of p_i',

$$S = \{(x, x) \mid x \in \mathsf{var}(p_i)\} \cup \{(x, x') \mid x \in \mathsf{var}(p_i)\}$$

is an $\mathcal{ELI}$-simulation from $\mathcal{A}_{p_i}$ to $\mathcal{A}_{p_i'}$ with $(x_0, x_0) \in S$. Therefore, by Lemma 14, $\mathcal{A}_{p_i'}, \mathcal{O} \models q_T(x_0)$ and $p_i' \subseteq_{\mathcal{O}} q_T$. Hence, by Point 1 of Lemma 13, $p_{i+1} \subseteq_{\mathcal{O}} q_T$ for $p_{i+1} = \mathsf{minimize}(p_i')$.

For Point 2, define a homomorphism g from $\mathsf{var}(p_{i+1})$ to $\mathsf{var}(p_i)$ by setting $g(x) = x$ for all $x \in \mathsf{var}(p_i) \cap \mathsf{var}(p_{i+1})$ and $g(x') = x$ for all $x' \in \mathsf{var}(p_{i+1}) \setminus \mathsf{var}(p_i)$. To establish that $|\mathsf{var}(p_{i+1})| > |\mathsf{var}(p_i)|$, we 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.

Claim 1. g is surjective.

Proof of Claim 1. Suppose that g is not surjective. Then some $y \in \mathsf{var}(p_i)$ does not occur in the image of g. Choose some $r(x, y) \in p_i$ and define $p_i^- = p_i \setminus \{r(x, y)\}$. Clearly, g is still a homomorphism from p_{i+1} to p_i^-. By Lemma 10, we can extend g to be a homomorphism from $\mathcal{U}_{p_{i+1},\mathcal{O}}$ to $\mathcal{U}_{p_i^-,\mathcal{O}}$. By Point 1, there is a homomorphism h from q_T to $\mathcal{U}_{p_{i+1},\mathcal{O}}$ with $h(x_0) = x_0$. Composing h and g yields a homomorphism g' from q_T to $\mathcal{U}_{p_i^-,\mathcal{O}}$ with $g'(x_0) = x_0$. Thus, $\mathcal{A}_{p_i^-}, \mathcal{O} \models q_T(x_0)$ which is in contradiction to the fact that p_i is obtained by applying minimize, and that this operation would replace p_i with p_i^-.

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

Proof of Claim 2. Suppose to the contrary that there is an $r(x, y) \in p_i$ with $r(g^-(x), g^-(y)) \notin p_{i+1}$. Then g is also a homomorphism from p_{i+1} to $p_i \setminus \{r(x, y)\}$ and using the same composition-of-homomorphisms argument as in the proof of Claim 1, we find a homomorphism h from q_T to $\mathcal{U}_{p_i \setminus \{r(x,y)\},\mathcal{O}}$ with $h(x_0) = x_0$. Hence $p_i \setminus \{r(x, y)\} \subseteq_{\mathcal{O}} q_T$. This contradicts the fact that $p_i = \mathsf{minimize}(p_i')$.

Claim 3. g is not injective.

Proof of Claim 3. Let $R_1(x_1, x_2), \ldots, R_n(x_n, x_1)$ be a cycle in p_i and $R_n(x_n, x_1)$ the atom that was chosen in the cycle doubling operation. 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. This contradicts the construction of p_{i+1} that removes $R(x_n, x_1)$.

For the induction step, let $j \geq 1$. By the induction hypothesis $x_j \notin \mathsf{var}(p_{i+1})$ and thus $x_j' \in \mathsf{var}(p_{i+1})$. Assume to the contrary of what is shown that $x_{j+1} \in \mathsf{var}(p_{i+1})$. Then $g(x_{j+1}) = x_{j+1}$, $g(x_j') = x_j$ 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. This contradicts the construction of p_{i+1}.

This completes the proof of Claim 3 and thus Point 2 of the lemma.

We show Point 3 of the lemma by induction on i. In the induction start, $\mathcal{O}$-saturatedness and satisfiability of p_1 w.r.t. $\mathcal{O}$ follows from $\mathcal{O}$-saturatedness and satisfiability of q w.r.t. $\mathcal{O}$ and the fact that minimize preserves those properties.

Now let p_i be $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$. Then p_i' is also $\mathcal{O}$ saturated, by construction. Moreover, we can construct a model $\mathcal{I}$ of $\mathcal{A}_{p_i'}$ and $\mathcal{O}$ by starting with $\mathcal{I} = \mathcal{A}_{p_i'}$ and attaching the trace subtrees below every $x \in \mathsf{ind}(\mathcal{A}_{p_i})$ in $\mathcal{U}_{\mathcal{A}_{p_i},\mathcal{O}}$ to x and x' in $\mathcal{I}$, and then adding (d, e) to $r^{\mathcal{I}}$ whenever $(d, e) \in s^{\mathcal{I}}$ and $\mathcal{O} \models s \sqsubseteq r$. The resulting interpretation $\mathcal{I}$ is a model of $\mathcal{A}_{p_i'}$ and $\mathcal{O}$. In particular, $\mathcal{I}$ satisfies all functionality assertions in $\mathcal{O}$, because every element in $\Delta^{\mathcal{I}}$ has the same number of r-successors and r-predecessors as its corresponding original element in $\mathcal{U}_{\mathcal{A}_{p_i},\mathcal{O}}$.

Since minimize preserves $\mathcal{O}$-saturatedness and satisfiability w.r.t. $\mathcal{O}$, p_{i+1} is therefore also $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$. ❑

Point 2 of Lemma 15 and Point 2 of Lemma 13 imply that treeify terminates and thus eliminates all cycles in q while maintaining $q \subseteq_{\mathcal{O}} q_T$. The next lemma makes this precise.

Lemma 16. *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. Moreover,* $\mathsf{treeify}(q)$ *runs in time polynomial in $|\mathsf{var}(q_T)| + ||q|| + ||\mathcal{O}||$.*

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 minimize. Thus, by Point 2 of Lemma 13, $|\mathsf{var}(p_i)| \leq |\mathsf{var}(q_T)|$ for all $i \geq i$. But by Point 2 of Lemma 15, p_{i+1} has more variables than p_i, for every i. It follows that the length n of the sequence of queries is at most $|\mathsf{var}(q_T)|$ and thus treeify stops at $p_n = \mathsf{treeify}(q)$. It also follows that p_n does not contain a cycle. Moreover, p_n is obtained by applying minimize and thus connected due to Point 3 of Lemma 13. Thus, p_n is an ELIQ.

It remains to argue that the running time is as claimed in the lemma. A cycle in p_i can be identified in time polynomial in $|\mathsf{var}(p_i)| \leq |\mathsf{var}(q_T)|$. Moreover, each call $\mathsf{minimize}(p'_i)$ makes at most $||p'_i||$ membership queries. It suffices to note that $||p'_i|| \leq 2 \cdot ||p_i||$ and that $||p_i|| \leq ||\mathcal{O}|| \cdot ||q_T||$ since $\mathsf{var}(p_i) \leq \mathsf{var}(q_T)$ and p_i uses only symbols from $\mathcal{O}$. ❑

We now analyze the main part of the algorithm. To this end, let $q_1(x_0), q_2(x_0), \ldots$ be the sequences of hypotheses q_H that the algorithm constructs. The following lemma summarizes their most important properties.

Lemma 17. *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. Point 1 is proved by induction on i. For $i = 1$, $q_1 = \mathsf{treeify}(q_H^0)$ and $q_H^0 \subseteq_{\mathcal{O}} q_T$ imply $q_1 \subseteq_{\mathcal{O}} q_T$ by Point 3 of Lemma 15. 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 Point 1 of Lemma 13.

Point 2 follows from the fact that $q_{i+1} = \mathsf{minimize}(q_F)$ for q_F some element of a frontier of q_i w.r.t. $\mathcal{O}$. By definition of frontiers and Point 1 of Lemma 13 it follows that $q_i \subseteq_{\mathcal{O}} q_{i+1}$ and $q_{i+1} \not\subseteq_{\mathcal{O}} q_i$, as required.

For Point 3, let h be a homomorphism from q_{i+1} to $\mathcal{U}_{q_i,\mathcal{O}}$ with $h(x_0) = x_0$. Assume to the contrary of what is to be shown that some $y \in \mathsf{var}(q_i)$ does not occur in the image of h. Choose some $r(x, y) \in q_i$ such that the distance from x_0 to x is strictly smaller than that from x_0 to y and let q_i^- be the restriction of $q_i \setminus \{r(x, y)\}$ to the atoms that contain only variables reachable from x_0 in $q_i \setminus \{r(x, y)\}$. We can argue as in the proof of Lemma 13 that h is a homomorphism from q_{i+1} to $\mathcal{U}_{q_i^-,\mathcal{O}}$. By Lemma 10, we can extend h to be a homomorphism from $\mathcal{U}_{q_{i+1},\mathcal{O}}$ to $\mathcal{U}_{q_i^-,\mathcal{O}}$. By Point 1, there is a homomorphism g from q_T to $\mathcal{U}_{q_{i+1},\mathcal{O}}$ with $h(x_0) = x_0$. Composing g and h yields a homomorphism g' from q_T to $\mathcal{U}_{q_i^-,\mathcal{O}}$ with $g'(x_0) = x_0$. Thus $\mathcal{A}_{q_i^-}, \mathcal{O} \models q_T(x_0)$ which is in contradiction to the fact that q_i is obtained by applying minimize, and that this operation would replace q_i with q_i^-. ❑

It remains to show that the algorithm terminates after polynomially many steps.

Lemma 18. *$q_n \equiv q_T$ for some $n \leq p(|\mathsf{var}(q_T)| + ||\mathcal{O}||)$, with p a polynomial.*

Proof. Every ELIQ q_i, with $i \geq 1$, is $\mathcal{O}$-saturated and satisfiable w.r.t. $\mathcal{O}$. This is easy to prove by induction on i. The induction start follows from the fact that the seed CQ is satisfiable w.r.t. $\mathcal{O}$ and $\mathcal{O}$-saturated, and from Point 3 of Lemma 15. The induction step follows from Point 3 of Lemma 13.

Point 2 of Lemma 13 thus yields $|\mathsf{var}(q_i)| \leq |\mathsf{var}(q_T)|$ for all $i \geq 1$. Moreover, Point 3 of Lemma 17 implies that $|\mathsf{var}(q_i)| \leq |\mathsf{var}(q_{i+1})|$. Hence, it remains to show that the length of any subsequence $q_j, \ldots, q_k$ with $|\mathsf{var}(q_j)| = \cdots = |\mathsf{var}(q_k)|$ is bounded by a polynomial in $\mathsf{var}(q_T) + ||\mathcal{O}||$.

Let h_ℓ for $\ell \in \{j, \ldots, k-1\}$ be the homomorphism from $q_{\ell+1}$ to $\mathcal{U}_{q_\ell,\mathcal{O}}$ that exists due to Point 2 of Lemma 17. Since $|\mathsf{var}(q_\ell)| = |\mathsf{var}(q_{\ell+1})|$ and by Point 3 of Lemma 17, h_ℓ is a bijection between $\mathsf{var}(q_{\ell+1})$ and $\mathsf{var}(q_\ell)$. Also by Point 2 of Lemma 17, h^- is not a homomorphism from q_ℓ to $\mathcal{U}_{q_{\ell+1},\mathcal{O}}$.

Therefore, one of the following two cases applies:

1. there is a concept atom $A(x_1) \in q_\ell$ such that $A(h_\ell^-(x_1)) \notin \mathcal{U}_{q_{\ell+1},\mathcal{O}}$ or
2. there is a role atom $r(x_1, x_2) \in q_\ell$ such that $r(h_\ell^-(x_1), h_\ell^-(x_2)) \notin \mathcal{U}_{q_{\ell+1},\mathcal{O}}$.

We show that each case can occur at most polynomially often in $\mathsf{var}(q_T) + ||\mathcal{O}||$.

We start with Case 1. It follows from the fact that h is a homomorphism that whenever $A(h_\ell^-(x_1)) \in q_{\ell+1}$, then $A(x_1) \in \mathcal{U}_{q_\ell,\mathcal{O}}$. Since q_ℓ is $\mathcal{O}$-saturated, this implies $A(x_1) \in q_\ell$. Moreover, $A(h_\ell^-(x_1)) \notin \mathcal{U}_{q_{\ell+1},\mathcal{O}}$ implies $A(h_\ell^-(x_1)) \notin q_{\ell+1}$. Consequently, q_ℓ contains at least one concept atom more than $q_{\ell+1}$. Thus, Case 1 can occur as most as often as there are concept atoms in q_1, and this number is bounded by $|\mathsf{var}(q_T)| \cdot ||\mathcal{O}||$ since $|\mathsf{var}(q_1)| \leq |\mathsf{var}(q_T)|$ and q_1 may only use symbols from $\mathcal{O}$.

In Case 2, consider the unique role atom $s(h_\ell^-(x_1), h_\ell^-(x_2)) \in q_{\ell+1}$. Since h_ℓ is a homomorphism, $s(x_1, x_2) \in \mathcal{U}_{q_\ell,\mathcal{O}}$. From $r(x_1, x_2) \in q_\ell$ and the construction of universal models, it follows that $\mathcal{O} \models r \sqsubseteq s$. From $r(h_\ell^-(x_1), h_\ell^-(x_2)) \notin \mathcal{U}_{q_{\ell+1},\mathcal{O}}$, it follows that $\mathcal{O} \not\models s \sqsubseteq r$. Thus, Case 2 can occur at most $||\mathcal{O}||$ times for each role atom in q_1. Since q_1 is a tree, $|\mathsf{var}(q_1)| \leq |\mathsf{var}(q_T)|$, and q_1 may only use symbols from $\mathcal{O}$, the number of such atoms is bounded by $|\mathsf{var}(q_T)| \cdot ||\mathcal{O}||$. ❑

Theorem 7. *AQ^s are not learnable under disjointness ontologies using only polynomially many 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 [Angluin, 1987b]. Essentially the same proof is given in [Funk *et al.*, 2021] 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

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

Note that S_n is a frontier of $\bot$ w.r.t. $\mathcal{O}_n$, if only AQ^ 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.[3]

Assume to the contrary of what is to be shown that AQ$^\wedge$ queries are learnable under disjointness ontologies using only polynomially many 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. ❑

Theorem 8. *ELIQs are not learnable under DL-Lite$^\mathcal{F}$ ontologies using only membership queries.*

Proof. We use the same ontology $\mathcal{O}$ as in the proof of Theorem 3, that is,

$$\mathcal{O} = \{\ A \sqsubseteq \exists r, \quad \exists r^- \sqsubseteq \exists r, \quad \exists r \sqsubseteq \exists s, \quad \mathsf{func}(r^-)\ \}.$$

To show that ELIQs are not learnable under $\mathcal{O}$ using only membership queries we use an infinite set $\mathcal{H}$ of hypotheses (i.e., candidate target queries) that cannot be distinguished by a finite number of membership queries. Let

$$\mathcal{H} = \{q^*\} \cup \{q_n \mid n \text{ prime}\},$$

where $q^*(x_1) = \{A(x_0), r(x_0, x_1), A(x_1)\}$ and each q_n is defined as follows:

$$\begin{aligned} q_n(x_1) = \{\ & A(x_0), r(x_0, x_1), r(x_1, x_2), \ldots, r(x_{n-1}, x_n), \\ & s(x_n, y), s(x'_n, y), \\ & r(x'_1, x'_2), \ldots, r(x'_{n-1}, x'_n), A(x'_1)\ \}. \end{aligned}$$

It is important to note that $q^* \subseteq_\mathcal{O} q_n$ for all $n \geq 1$. Intuitively, this makes it impossible for the learner to distinguish between q^* being the target query and one of the q_n being the target query. If, for example, the learner asks a membership query '$\mathcal{A}^*, \mathcal{O} \models q_T(a)$?', where $\mathcal{A}^*$ is q^* viewed as an ABox, then the oracle will answer 'yes' and the learner has not gained any additional information.

Now, the strategy of the oracle to answer a membership query '$\mathcal{A}, \mathcal{O} \models q_T(a)$?' is as follows:

1. if $\mathcal{A}, \mathcal{O} \models q^*(a)$, then reply "yes";
2. otherwise, reply "no" and remove from $\mathcal{H}$ any q that satisfies $\mathcal{A}, \mathcal{O} \models q(a)$.

An important aspect of this strategy is that, as proved below, only finitely many hypotheses q are removed whenever Case 2 above applies. Consequently, after any number of membership queries, the set of remaining hypotheses $\mathcal{H}$ is infinite and contains q^*. The learner can then, however, not distinguish between q^* and the remaining hypotheses and thus not identify the target query. In particular, the presence of $q^* \in \mathcal{H}$ prevents the learner from simply going through all $q_i \in \mathcal{H}$, asking membership queries with ABoxes that take the form of these queries, and identifying q_i as the target query when the membership query for q_i succeeds.

Claim. Let $\mathcal{A}$ an ABox and $a \in \mathsf{ind}(\mathcal{A})$. If $\mathcal{A}, \mathcal{O} \not\models q^*(a)$, then $\mathcal{A}, \mathcal{O} \models q_n(a)$ for only finitely many primes n.

Proof of the claim. Let $\mathcal{A}$ an ABox and $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A}, \mathcal{O} \not\models q^*(a)$. Then $\mathcal{A}$ satisfies $\mathsf{func}(r^-)$. Suppose to the contrary of what we have to show that there are infinitely many primes n such that $\mathcal{A}, \mathcal{O} \models q_n(a)$ and let h_n be the witnessing homomorphisms from q_n to $\mathcal{U}_{\mathcal{A},\mathcal{O}}$ with $h(x_1) = a$.

- If $h_n(x_n) \notin \mathsf{ind}(\mathcal{A})$ for some prime n with $\mathcal{A}, \mathcal{O} \models q_n(a)$, then $h_n(x'_n) = h_n(x_n)$ due to the tree structure of the non-ABox part of $\mathcal{U}_{\mathcal{A},\mathcal{O}}$. Since r^- is functional, it follows that $h_n(x'_j) = h_n(x'_j)$ for all j with $1 \leq j \leq n$. Since $A(x'_1) \in q_n$, also $A(h_n(x'_1)) = A(h_n(x_1)) = A(a) \in \mathcal{A}$. Since also $A(x_0), r(x_0, x_1) \in q_n$, we have $A(h_n(x_0)), r(h_n(x_0), h_n(x_1)) \in \mathcal{U}_{\mathcal{A},\mathcal{O}}$, and thus h_n is a homomorphism from q^* to $\mathcal{U}_{\mathcal{A},\mathcal{O}}$ with $h_n(x_1) = a$. Hence, $\mathcal{A}, \mathcal{O} \models q^*(a)$, a contradiction.
- Otherwise, $h_n(x_n) \in \mathsf{ind}(\mathcal{A})$ for all primes n with $\mathcal{A}, \mathcal{O} \models q_n(a)$. Since $\mathcal{A}$ is finite, there is an element $b \in \mathsf{ind}(\mathcal{A})$ such that $h_m(x_m) = b$ for infinitely many primes m. Thus, there is an r-path of length m from a to b in $\mathcal{A}$ for infinitely many primes m. Since $\mathcal{A}$ is finite and satisfies $\mathsf{func}(r^-)$, this is only possible if $a = b$, $r(a, a) \in \mathcal{A}$, and $h_m(x_j) = a$ for all considered m and all j with $1 \leq j \leq m$. Since also $A(x_0), r(x_0, x_1) \in q_n$ and $\mathcal{A}$ satisfies $\mathsf{func}(r^-)$, we further have $h_n(x_0) = a$ for all primes n with $\mathcal{A}, \mathcal{O} \models q_n(a)$ and $A(a) \in \mathcal{A}$. But then $\mathcal{A}, \mathcal{O} \models q^*(a)$, a contradiction.

❑

[3] 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.