TEMPORALISING UNIQUE CHARACTERISABILITY AND LEARNABILITY OF ONTOLOGY-MEDIATED QUERIES

TECHNICAL REPORT

Jean Christoph Jung
TU Dortmund University, UK
jean.jung@tu-dortmund.de

Vladislav Ryzhikov
Birkbeck, University of London, UK
vlad@dcs.bbk.ac.uk

Frank Wolter
University of Liverpool, UK
wolter@liverpool.ac.uk

Michael Zakharyaschev
Birkbeck, University of London, UK
michael@dcs.bbk.ac.uk

June 13, 2023

ABSTRACT

Recently, the study of the unique characterisability and learnability of database queries by means of examples has been extended to ontology-mediated queries. Here, we study in how far the obtained results can be lifted to temporalised ontology-mediated queries. We provide a systematic introduction to the relevant approaches in the non-temporal case and then show general transfer results pinpointing under which conditions existing results can be lifted to temporalised queries.

1 Introduction

Temporal ontology-mediated query answering provides a framework for accessing temporal data using background knowledge in the form of a logical theory, often called an ontology. Within this framework, queries are usually constructed by combining a domain query language such as conjunctive queries (CQs) with linear temporal logic (*LTL*) operators. Ontologies range from pure description logic (DL) ontologies [11, 16], which hold at all timepoints, to suitable combinations of DLs with *LTL* [7, 9, 28, 10, 48]. On the query side, *LTL* is combined with domain queries formulated usually as conjunctive queries (CQs). To ensure good semantic and computational properties, one often restricts the use of *LTL* operators to monotone operators and the domain queries to acyclic CQs such as the class ELIQ of CQs that are equivalent to $\mathcal{ELI}$-queries. On the ontology side, the most popular languages are based on the *DL-Lite* family of DLs, which underpins the OWL2 DL profile of the W3C standard for ontology-based data access [17, 6].

Our aim in this paper is *(i)* to find out in how far temporal queries can be (polynomially) characterised using temporal data instances if ontologies encoding background knowledge are present and *(ii)* to apply the results to study the (polynomial) learnability of temporal queries in Angluin's framework of exact learning [4]. Investigating the computation, size, and shape of unique characterisations of queries by data examples contributes to the *query-by-example* paradigm in databases [37] as the data examples can be naturally used to illustrate, explain, and construct queries. Unique characterisations also provide a 'non-procedural' necessary condition for (polynomial time) exact learnability using membership queries, where membership queries to the oracle take the form 'does $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}_T(a)$ hold?' and thus provide access to query answers of the target query $\boldsymbol{q}_T$ to be learned. We focus our investigation on temporal extensions of the class ELIQ with operators $\bigcirc$ ('in the next time point'), $\Diamond$ ('at some point in the future'), and U ('until').

Recently, significant progress has been made in understanding the unique characterisability and learnability of non-temporal ELIQs mediated by DL ontologies [44, 26]. Also, rather general results have been obtained about the characterisation and learnability of temporal queries, but so far without ontologies [22]. From a technical viewpoint, in this paper we combine these two directions to study the temporalisation of unique characterisability and learnability under DL ontologies.

We next provide an overview of our results. Let $\mathcal{O}$ be an ontology and $\mathcal{Q}$ a class of conjunctive queries, which we assume for simplicity to have a single answer variable. We say that a query $\boldsymbol{q} \in \mathcal{Q}$ *fits* a pair $E = (E^+, E^-)$ of finite sets E^+ and E^- of pointed data instances $(\mathcal{D}, a)$ wrt $\mathcal{O}$ if $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}(a)$ for all $(\mathcal{D}, a) \in E^+$, and $\mathcal{O}, \mathcal{D} \not\models \boldsymbol{q}(a)$ for all $(\mathcal{D}, a) \in E^-$. Then E *uniquely characterises* $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$ if $\boldsymbol{q}$ is the only (up to equivalence modulo $\mathcal{O}$) query in $\mathcal{Q}$ that fits E wrt $\mathcal{O}$. An ontology language $\mathcal{L}$ *admits (polysize) characterisations within* $\mathcal{Q}$ if every $\boldsymbol{q} \in \mathcal{Q}$ has a (polysize) characterisation wrt to any $\mathcal{L}$-ontology within $\mathcal{Q}$.

Non-temporal case. We begin by summarising the relevant results that will be used as a black box in our investigation of temporalised queries. Let $\mathcal{O}$ be an FO-ontology (typically in some DL). Given queries $\boldsymbol{q}_1$, $\boldsymbol{q}_2$, we write $\boldsymbol{q}_1 \models_{\mathcal{O}} \boldsymbol{q}_2$ and say that $\boldsymbol{q}_1$ is *contained* in $\boldsymbol{q}_2$ wrt $\mathcal{O}$ if $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}_1(a)$ implies $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}_2(a)$, for any pointed data instance $\mathcal{D}, a$. We utilise a well-known reduction of containment to query entailment. An ontology $\mathcal{O}$ *admits containment reduction* if, for any CQ $\boldsymbol{q}(x)$, there is a pointed data instance $(\hat{\boldsymbol{q}}, a)$ such that $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}'$ iff $\mathcal{O}, \hat{\boldsymbol{q}} \models \boldsymbol{q}'(a)$, for any CQ $\boldsymbol{q}'(x)$ (we also demand a few minor technical conditions given in teh paper). An ontology language $\mathcal{L}$ *admits containment reduction* if every $\mathcal{L}$-ontology does. It is easy to see that if $\mathcal{L}$ admits both (polysize) unique characterisations within $\mathcal{Q}$ and containment reduction, then every $\boldsymbol{q} \in \mathcal{Q}$ has a (polysize) *singular*$^+$*characterisation* E within $\mathcal{Q}$ with $E^+ = \{\hat{\boldsymbol{q}}\}$. Containment reduction is a rather general condition (unfortunately not shared by temporalisations, see below): FO without equality including DLs such as $\mathcal{ALCHI}$ and *DL-Lite*$_{\mathcal{H}}$ [17] (aka *DL-Lite*$^{\mathcal{H}}_{core}$ [6]) and also some DLs with limited counting such as *DL-Lite*$_{\mathcal{F}}$ [17] (aka *DL-Lite*$^{\mathcal{F}}_{core}$ [6]) admit containment reduction but $\mathcal{ALCQ}$ does not.

The two main approaches to compute E^- and obtain singular$^+$characterisations for languages with containment reduction are based on frontiers and splittings (aka dualities) [44]. A *frontier of* $\boldsymbol{q}$ *wrt* $\mathcal{O}$ *within* $\mathcal{Q}$ is any set $\mathcal{F}_{\boldsymbol{q}} \subseteq \mathcal{Q}$ such that (a) $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}'$ and $\boldsymbol{q}' \not\models_{\mathcal{O}} \boldsymbol{q}$, for all $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$; and (b) if $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}''$ for $\boldsymbol{q}'' \in \mathcal{Q}$, then $\boldsymbol{q}'' \models_{\mathcal{O}} \boldsymbol{q}$ or there is $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$ with $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}''$. An ontology language $\mathcal{L}$ *admits* (*polysize*) *finite frontiers within* $\mathcal{Q}$ if every $\boldsymbol{q} \in \mathcal{Q}$ has a (polysize) finite frontier wrt to any $\mathcal{L}$-ontology within $\mathcal{Q}$. The following theorem summarises what is currently known about frontiers of ELIQs. While (i) and (ii) are shown in [26], we show (iii) in this paper.

Theorem 1. (i) *DL-Lite*$_{\mathcal{H}}$ *and the fragment DL-Lite*$^{-}_{\mathcal{F}}$ *of DL-Lite*$_{\mathcal{F}}$*, in which R^- is not functional for any $B \sqsubseteq \exists R$, admit polysize frontiers within ELIQ.* (ii) *DL-Lite*$_{\mathcal{F}}$ *does not admit finite frontiers within ELIQ.* (iii) *$\mathcal{EL}$ does not admit finite frontiers within ELIQ.*

The frontier of a query supplies the negative examples for a singular$^+$unique characterisation.

Theorem 2. *If $\mathcal{L}$ admits both* (*polysize*) *frontiers within $\mathcal{Q}$ and containment reduction, then $\mathcal{L}$ admits* (*polysize*) *singular*$^+$*characterisations within $\mathcal{Q}$, with $E^- = \mathcal{F}_{\boldsymbol{q}}$, for any $\boldsymbol{q} \in \mathcal{Q}$.*

The second path to singular$^+$characterisations is via finite splittings, which only exist if a finite signature σ of predicates is fixed. Let $\mathcal{Q}$ be a class of queries and $\mathcal{Q}^\sigma$ its restriction to σ, $Q \subseteq \mathcal{Q}^\sigma$ finite, and $\mathcal{O}$ a σ-ontology. A set $\mathcal{S}(Q)$ of pointed σ-data instances $(\mathcal{A}, a)$ is called a *split-partner* for Q wrt $\mathcal{O}$ within $\mathcal{Q}^\sigma$ if, for all $\boldsymbol{q}' \in \mathcal{Q}^\sigma$, we have $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}'(a)$ for some $(\mathcal{A}, a) \in \mathcal{S}(Q)$ iff $\boldsymbol{q}' \not\models_{\mathcal{O}} \boldsymbol{q}$ for all $\boldsymbol{q} \in Q$. An ontology language $\mathcal{L}$ *has general split-partners within* $\mathcal{Q}^\sigma$ if all finite sets of $\mathcal{Q}^\sigma$-queries have split partners wrt any σ-ontology in $\mathcal{L}$. We show the following results.

Theorem 3. (i) *$\mathcal{ALCHI}$ has exponential-size general split-partners within σ-ELIQ,* (ii) *even wrt to the empty ontology, no polysize split-partners exit within σ-ELIQ [22].*

Thus, $\mathcal{EL}$ has finite general split-partners but no frontiers within ELIQ, and *DL-Lite*$^{-}_{\mathcal{F}}$ has finite frontiers but no finite general split-partners within ELIQ. This is in contrast to the ontology-free case where frontiers and splittings are more closely linked [44]. One can now show the following analogue of Theorem 2 for splittings:

Theorem 4. *If $\mathcal{L}$ admits* (*polysize*) *general split-partners within $\mathcal{Q}^\sigma$ and containment reduction, then $\mathcal{L}$ admits* (*polysize*) *singular*$^+$*characterisations within $\mathcal{Q}$, with $E^- = \mathcal{S}(\{\boldsymbol{q}\})$, for any $\boldsymbol{q} \in \mathcal{Q}^\sigma$.*

Temporalisation. A *temporal data instance* is a sequence $\mathcal{A}_0, \dots, \mathcal{A}_n$ of domain data instances $\mathcal{A}_i$ with i regarded as a *timestamp*. To query temporal data, we equip standard CQs with the operators of linear temporal logic *LTL*. Within this framework, various query languages that admit (polysize) unique characterisations and learnability have been identified in the case when no background ontology is present [22]. Here, we assume that the temporal data is mediated by a standard (atemporal) DL ontology whose axioms are supposed to be true at all times. We consider a few families of temporal queries defined in [22] that are built from domain queries in a given class $\mathcal{Q}$ (say, ELIQ or conjunctions of concept names, denoted $\mathcal{P}$) using $\wedge$ and the temporal operators $\bigcirc$ (at the next moment), $\Diamond$ (some time later), $\Diamond_r$ (now or later), and U (strict until): the family $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ of *path queries* of the form

$$\boldsymbol{q} = \boldsymbol{r}_0 \wedge \boldsymbol{o}_1(\boldsymbol{r}_1 \wedge \boldsymbol{o}_2(\boldsymbol{r}_2 \wedge \cdots \wedge \boldsymbol{o}_n \boldsymbol{r}_n)),$$

where $\boldsymbol{o}_i \in \{\bigcirc, \Diamond, \Diamond_r\}$ and $\boldsymbol{r}_i \in \mathcal{Q}$; the family $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{Q}^\sigma)$ of *path queries*

$$\boldsymbol{q} = \boldsymbol{r}_0 \wedge (\boldsymbol{l}_1 \mathbin{\mathsf{U}} (\boldsymbol{r}_1 \wedge (\boldsymbol{l}_2 \mathbin{\mathsf{U}} (\dots (\boldsymbol{l}_n \mathbin{\mathsf{U}} \boldsymbol{r}_n) \dots)))),$$

where $\boldsymbol{r}_i \in \mathcal{Q}^\sigma$, $\boldsymbol{l}_i \in \mathcal{Q}^\sigma \cup \{\bot\}$; and its subfamily $\mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{Q}^\sigma)$ of *peerless queries* in which $\boldsymbol{r}_i \not\models_{\mathcal{O}} \boldsymbol{l}_i$ and $\boldsymbol{l}_i \not\models_{\mathcal{O}} \boldsymbol{r}_i$. The subfamily $\mathit{LTL}_p^{\bigcirc\Diamond}(\mathcal{Q})$ restricts $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ to the operators $\bigcirc$ and $\Diamond$; note that $\Diamond \boldsymbol{q} \equiv \bigcirc\Diamond_r \boldsymbol{q}$.

Temporal queries have a few essential differences from the domain ones. First, no example set can distinguish the query $\Diamond_r(A \wedge B)$ from $\Diamond_r(A \wedge (\Diamond_r B \wedge \Diamond_r(A \wedge \dots)))$ with sufficiently many alternating A, B. A syntactic criterion (excluding proper conjunctions that do not have a $\bigcirc$-neighbour) of unique characterisability of queries in $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{P})$, called *safety*, was found in [22]. Second, containment reduction does not work any more since to characterise, say, $\Diamond A$ *two* positive examples are needed. By generalising safety in a natural way, we obtain our first transfer result:

Theorem 5. *Let $\mathcal{L}$ admit (polysize) singular$^+$characterisations within $\mathcal{Q}$ and $\mathcal{O}$ be a $\mathcal{L}$-ontology that admits containement reduction. Then*

(i) *$\boldsymbol{q} \in \mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ is (polysize) uniquely characterisable wrt $\mathcal{O}$ within $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ iff $\boldsymbol{q}$ is safe wrt $\mathcal{O}$;*

(ii) *all $\boldsymbol{q} \in \mathit{LTL}_p^{\bigcirc\Diamond}(\mathcal{Q})$ are (polysize) uniquely characterisable wrt $\mathcal{O}$.*

(iii) *If $\mathcal{O}$ admits polysize singular$^+$characterisations within $\mathcal{Q}$, then $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ is polynomially characterisable for bounded temporal depth.*

As a consequence of the above results, we obtain, e.g., that every safe query in $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\text{ELIQ})$ is polynomially characterisable wrt any *DL-Lite*$_{\mathcal{H}}$ or *DL-Lite*$_{\mathcal{F}}^-$ ontology and exponentially characterisable wrt any $\mathcal{ALCHI}$-ontology. Our second transfer result is as follows:

Theorem 6. *Let $\mathcal{L}$ have (exponential-size) general split-partners within $\mathcal{Q}^\sigma$ and let $\mathcal{O}$ be a σ-ontology in $\mathcal{L}$ that admits containment reduction. Then every $\boldsymbol{q} \in \mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{Q}^\sigma)$ is (exponential-size) uniquely characterisable within $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{Q}^\sigma)$.*

As a consequence, we obtain that every query in $\mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{Q}^\sigma)$, where $\mathcal{Q}^\sigma$ is the class of σ-ELIQs, is exponentially uniquely characterisable within $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{Q}^\sigma)$ wrt any $\mathcal{ALCHI}$ ontology.

Learning. We apply our results on characterisability to learnability of queries in $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\text{ELIQ})$ wrt ontologies in Angluin's framework of exact learning [4]. In the non-temporal case, exact learning of queries has recently been studied [44, 24, 26, 25]. Given some class $\mathcal{Q}$ of queries and an ontology $\mathcal{O}$, the *learner* aims to identify a *target query* $\boldsymbol{q}_T \in \mathcal{Q}$ using membership queries of the form 'does $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}(a)$ hold?' to the *teacher*. It is assumed that the target query $\boldsymbol{q}_T$ uses only symbols that occur in the ontology $\mathcal{O}$. We call $\mathcal{Q}$ *polynomial query* (*polynomial-time*) *learnable wrt $\mathcal{L}$-ontologies using membership queries* if there is a learning algorithm that receives an $\mathcal{L}$-ontology $\mathcal{O}$ and an example $(\mathcal{D}, a)$ with $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}_T(a)$ with $\mathcal{D}$ satisfiable under $\mathcal{O}$, and constructs $\boldsymbol{q}_T$ (up to equivalence wrt $\mathcal{O}$) using polynomially-many queries of polynomial size (in time polynomial) in the size of $\boldsymbol{q}_T, \mathcal{O}, \mathcal{D}$.

As we always construct example sets effectively, our unique (exponential) characterisability results imply (exponential-time) learnability with membership queries. Obtaining polynomial-time learnability from polynomial characterisations is more challenging and, in fact, not always possible. We concentrate on ontologies formulated in fragments $\mathcal{L}$ of the DL $\mathcal{ELHIF}$ which are in normal form [12], but conjecture that our results continue to hold in general. *$\mathcal{L}$ admits polytime instance checking* if $\mathcal{O}, \mathcal{D} \models A(a)$, for a concept name A, can be decided in polynomial time. *Meet-reducibility* is in polytime if it can be checked in polytime whether an ELIQ is equivalent to a proper conjunction of ELIQs wrt to an $\mathcal{L}$-ontology. The following is shown by lifting the techniques developed for the non-temporal setting in [25, 26] to the temporal case:

Theorem 7. *Let $\mathcal{L}$ be an ontology language that contains only $\mathcal{ELHIF}$-ontologies in normal form and that admits polysize frontiers within ELIQ that can be computed. Then:*

(i) *The class of safe queries in $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathit{ELIQ})$ is polynomial query learnable wrt $\mathcal{L}$-ontologies using membership queries.*

(ii) *The class $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathit{ELIQ})$ is polynomial query learnable wrt $\mathcal{L}$-ontologies using membership queries if the learner knows the temporal depth of the target query in advance.*

(iii) *$\mathit{LTL}_p^{\bigcirc\Diamond}(\mathit{ELIQ})$ is polynomial query learnable wrt $\mathcal{L}$-ontologies using membership queries.*

If $\mathcal{L}$ further admits polynomial-time instance checking and polynomial-time computable frontiers within ELIQ, then in (ii) and (iii), polynomial query learnability can be replaced by polynomial-time learnability. If, in addition, meet-reducibility wrt $\mathcal{L}$-ontologies is in polynomial time, then also in (i) polynomial query learnability can be replaced by polynomial-time learnability.

Theorem 7 fully applies to *DL-Lite*$_{\mathcal{F}}^{-}$ as it enjoys all properties mentioned, while *DL-Lite*$_{\mathcal{H}}$ enjoys all properties mentioned except that meet-reducibility can be checked in poly-time. Most importantly, *DL-Lite*$_{\mathcal{F}}^{-}$ and *DL-Lite*$_{\mathcal{H}}$ admit polynomial time computable frontiers [26].

2 Related Work

Recently, the unique characterisation framework for temporal formulas introduced in [22] and underpinning this article has been generalised to allow for finitely representable transfinite words as examples in characterising sets [41]. The results are not directly applicable to the problems we are concerned with as the queries have no DL component and no ontology is present. We conjecture, however, that it is possible to extend the techniques used in [41] to the more general languages considered here.

Our work is closely related to work on exact learnability of finite automata and temporal logic formulas. In fact, the seminal paper by Angluin [3] has given rise to a large body of work on exact learning of (variations of) finite automata, for example, [42, 1, 19, 30]. This work has mainly focused on learning using a combination of membership queries with other powerful types of queries such as equivalence queries. The use of two or more types of queries is motivated by the fact that otherwise one cannot efficiently learn the respective automata models. The main difference between this article and work on automata learning is that we focus on queries for which the corresponding formal languages form only a small subset of the regular languages and this restriction enables us to focus on characterisability and learnability with membership queries.

There has hardly been any work directly concerned with exact learning of temporal formulas [18]. In contrast, passive learning of *LTL*-formulas has recently received significant attention; see [36, 39, 18, 21, 23].

In the database and KR communities, there has been extensive work on identifying queries and concept descriptions from data examples [43, 34, 45, 40, 44]. For instance, in reverse engineering of queries, the goal is typically to decide whether there is a query that fits (or separates) a set of positive and negative examples. Relevant work under the closed world assumption include [5, 13] and under the open world assumption [35, 29, 27, 31].

The use of unique characterisations to explain and construct schema mappings has been promoted and investigated by Kolaitis [33] and Alexe et al. [2].

Unique characterisability of modal logic formulas has recently been studied in [46]. In contrast to our work, this is done under the closed world assumption and without ontologies.

3 Preliminaries

We begin by introducing 'two-dimensional' query languages that combine instance queries over the object domain in the standard description logics $\mathcal{EL}$ and $\mathcal{ELI}$ [12] or more general conjunctive queries with temporal *LTL*-queries over the temporal domain [8]. Background knowledge over the object domain is assumed to be given by a standard one-dimensional description logic ontology whose axioms are supposed to be true at all times.

3.1 Ontologies and Queries over the Object Domain

By a *relational signature* we mean in this paper any non-empty finite set, σ, of unary and binary predicate symbols, typically denoted A, B and P, R, respectively. A σ-*data instance*, $\mathcal{A}$, is any non-empty finite set of *atoms* of the form $A(a)$ and $P(a, b)$ with $A, P \in \sigma$ and *individual names* (or *constants*) a, b, and also $\top(a)$, which simply says that the domain contains the individual a. We denote by *ind*$(\mathcal{A})$ the set of individual names in $\mathcal{A}$ and by P^- the *inverse* of P (with $P^{--} = P$), assuming that $P^-(a, b) \in \mathcal{A}$ iff $P(b, a) \in \mathcal{A}$. We assume that S ranges over binary predicates and their inverses. A *pointed data instance* is a pair $(\mathcal{A}, a)$ with $a \in$ *ind*$(\mathcal{A})$.

In general, an *ontology*, $\mathcal{O}$, is a finite set of first-order (FO) sentences over the given signature σ. Ontologies and data instances are interpreted in structures $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with domain $\Delta^{\mathcal{I}} \neq \emptyset$, $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ for unary A, $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ for binary P, and $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ for individual names a. As usual in database theory, we assume that $a^{\mathcal{I}} \neq b^{\mathcal{I}}$ for distinct a, b; moreover, to simplify notation, we adopt the *standard name assumption* and interpret each individual name by itself, i.e., take $a^{\mathcal{I}} = a$. Thus, $\mathcal{I}$ is a *model* of a data instance $\mathcal{A}$ if $a \in A^{\mathcal{I}}$ and $(a, b) \in P^{\mathcal{I}}$, for all $A(a) \in \mathcal{A}$ and $P(a, b) \in \mathcal{A}$ (remember that $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$). $\mathcal{I}$ is a *model* of an ontology $\mathcal{O}$ if all sentences in $\mathcal{O}$ are true in $\mathcal{I}$. $\mathcal{O}$ and $\mathcal{A}$ are *satisfiable* if they have a model.

The concrete ontology languages we deal with in this paper are certain members of the *DL-Lite* family, $\mathcal{ALCHI}$, $\mathcal{ELHIF}$, and some fragments thereof. For the reader's convenience, we define these languages below (in FO-terms):

DL-Lite$_\mathcal{F}$ [17] (aka *DL-Lite*$^\mathcal{F}_{core}$ [6]) allows axioms of the following forms:

$$\forall x\,(B(x) \to B'(x)), \quad \forall x\,(B(x) \land B'(x) \to \bot), \quad \forall x, y, z\,(S(x, y) \land S(x, z) \to (y = z)), \tag{1}$$

where *basic concepts* $B(x)$ are either $A(x)$ or $\exists y\, S(x, y)$. In the DL parlance, the first two axioms in (1) are written as $B \sqsubseteq B'$ and $B \sqcap B' \sqsubseteq \bot$ and called a *concept inclusion* (CI) and *concept disjointness constraint*, respectively. The third axiom, written $\geq 2\, S \sqsubseteq \bot$ or fun(S) in DL, states that relation S^- is *functional*, and so is called a *functionality constraint*.

DL-Lite$^-_\mathcal{F}$ [26] is a fragment of *DL-Lite*$_\mathcal{F}$, in which CIs $B \sqsubseteq B'$ cannot have $B' = \exists S$ with functional S^-.

DL-Lite$_\mathcal{H}$ [17] (aka *DL-Lite*$^\mathcal{H}_{core}$ [6]) is obtained from *DL-Lite*$_\mathcal{F}$ by disallowing the functionality constraint in (1) and adding axioms of the form

$$\forall x, y\,(S(x, y) \to S'(x, y)) \tag{2}$$

known as *role inclusions* and often written as $S \sqsubseteq S'$.

$\mathcal{ALCHI}$ [12] has the same role inclusions as in (2) but more expressive concept inclusions $\forall x\,(C_1(x) \to C_2(x))$, in which the *concepts* C_i are defined inductively starting from atoms $\top(x)$ (which is always true) and $A(x)$ and using the constructors $C(x) \land C'(x)$, $\neg C(x)$, and $\exists y\,(S(x, y) \land C(y))$, for a fresh y—or $C \sqcap C'$, $\neg C$, and $\exists S.C$ in DL terms.

$\mathcal{ELHIF}^\bot$ [12] has role inclusions (2), functionality constrains, and concept inclusions with concepts built from atoms and $\bot$ using $\land$ and $\exists y\,(S(x, y) \land C(y))$ only. $\mathcal{ELFI}^\bot$ is the fragment of $\mathcal{ELHIF}^\bot$ without role inclusions.

We reserve the character $\mathcal{L}$ to denote an ontology language.

The most general query language over the object domain we consider in this paper consists of *conjunctive queries* (CQs) $\boldsymbol{q}(x)$ over the given signature σ with a single *answer variable* x. We usually think of $\boldsymbol{q}(x)$ as the set of its atoms and denote by *var*$(\boldsymbol{q})$ the set of individual variables in $\boldsymbol{q}$. We say that $\boldsymbol{q}(x)$ is *satisfiable* wrt an ontology $\mathcal{O}$ if the set $\mathcal{O} \cup \{\boldsymbol{q}(x)\}$ has a model. The set of predicate symbols in $\boldsymbol{q}$ is denoted by *sig*$(\boldsymbol{q})$. The restriction of $\boldsymbol{q}(x)$ to a set V of variables is the CQ that is obtained from $\boldsymbol{q}(x)$ by omitting the atoms with a variable not in V.

Given a CQ $\boldsymbol{q}(x)$, an ontology $\mathcal{O}$, and a data instance $\mathcal{A}$, we say that $a \in$ *ind*$(\mathcal{A})$ is a (*certain*) *answer to* $\boldsymbol{q}$ *over* $\mathcal{A}$ *wrt* $\mathcal{O}$ and write $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}(a)$ if $\mathcal{I} \models \boldsymbol{q}(a)$ for all models $\mathcal{I}$ of $\mathcal{O}$ and $\mathcal{A}$. Recall that $\emptyset, \mathcal{A} \models \boldsymbol{q}(a)$ iff there is function $h\colon$ *var*$(\boldsymbol{q}) \to \mathcal{A}$ such that $h(x) = a$, $A(y) \in \boldsymbol{q}$ implies $A(h(y)) \in \mathcal{A}$, and $P(y, z) \in \boldsymbol{q}$ implies $P(h(y), h(z)) \in \mathcal{A}$. Such a function h is called a *homomorphism* from $\boldsymbol{q}$ to $\mathcal{A}$, written $h\colon \boldsymbol{q} \to \mathcal{A}$; h is *surjective* if $h(\mathit{var}(\boldsymbol{q})) = \mathit{ind}(\mathcal{A})$.

We say that a CQ $\boldsymbol{q}_1(x)$ is *contained* in a CQ $\boldsymbol{q}_2(x)$ wrt an ontology $\mathcal{O}$ and write $\boldsymbol{q}_1 \models_\mathcal{O} \boldsymbol{q}_2$ if $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}_1(a)$ implies $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}_2(a)$, for any data instance $\mathcal{A}$ and any $a \in \mathit{ind}(\mathcal{A})$. If $\boldsymbol{q}_1 \models_\mathcal{O} \boldsymbol{q}_2$ and $\boldsymbol{q}_2 \models_\mathcal{O} \boldsymbol{q}_1$, we say that $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ are *equivalent wrt* $\mathcal{O}$, writing $\boldsymbol{q}_1 \equiv_\mathcal{O} \boldsymbol{q}_2$. For $\mathcal{O} = \emptyset$, we often write $\boldsymbol{q}_1 \equiv \boldsymbol{q}_2$ instead of $\boldsymbol{q}_1 \equiv_\emptyset \boldsymbol{q}_2$. We call $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ *compatible wrt* $\mathcal{O}$ if $\boldsymbol{q}_1 \land \boldsymbol{q}_2$ is satisfiable wrt $\mathcal{O}$.

Two smaller query languages we need are $\mathcal{ELI}$*-queries* (or ELIQs, for short) that can be defined by the grammar

$$\boldsymbol{q} \quad := \quad \top \quad | \quad A \quad | \quad \exists S.\boldsymbol{q} \quad | \quad \boldsymbol{q} \land \boldsymbol{q}'$$

and $\mathcal{EL}$*-queries* (or ELQs), which are ELIQs without inverses P^-. Semantically, $\boldsymbol{q}$ has the same meaning as the *tree-shaped CQ* $\boldsymbol{q}(x)$ that is defined inductively starting from atoms $\top(x)$ and $A(x)$ and using the constructors $\exists y\,(S(x, y) \land \boldsymbol{q}(y))$, for a fresh y, and $\boldsymbol{q}(x) \land \boldsymbol{q}'(x)$. The only free (i.e., answer) variable in $\boldsymbol{q}$ is x. We reserve the character $\mathcal{Q}$ for denoting a class of queries with answer variable x such that whenever $\boldsymbol{q}_1, \boldsymbol{q}_2 \in \mathcal{Q}$, then $\boldsymbol{q}_1 \land \boldsymbol{q}_2 \in \mathcal{Q}$; by $\mathcal{Q}^\sigma$ we denote the class of those queries in $\mathcal{Q}$ that are built from predicates in σ. The classes of all σ-ELIQs and σ-ELQs are denoted by ELIQ$^\sigma$ and ELQ$^\sigma$, respectively.

It will be convenient to include the 'inconsistency query' $\bot(x)$ into all of our query classes. By definition, we have $\mathcal{O}, \mathcal{A} \models \bot(a)$ iff $\mathcal{O}$ and $\mathcal{A}$ are inconsistent.

We say that an ontology $\mathcal{O}$ *admits containment reduction* if, for any CQ $\boldsymbol{q}(x)$, there is a pointed data instance $(\hat{\boldsymbol{q}}, a)$ such that the following conditions hold:

(cr$_1$) $\boldsymbol{q}(x)$ is satisfiable wrt $\mathcal{O}$ iff $\mathcal{O}$ and $\hat{\boldsymbol{q}}$ are satisfiable;

(cr$_2$) there is a surjective homomorphism $h\colon \boldsymbol{q} \to \hat{\boldsymbol{q}}$ with $h(x) = a$;

(cr$_3$) if $\boldsymbol{q}(x)$ is satisfiable wrt $\mathcal{O}$, then for every CQ $\boldsymbol{q}'(x)$, we have $\boldsymbol{q} \models_\mathcal{O} \boldsymbol{q}'$ iff $\mathcal{O}, \hat{\boldsymbol{q}} \models \boldsymbol{q}'(a)$.

An ontology language $\mathcal{L}$ is said to *admit containment reduction* if every $\mathcal{L}$-ontology does. If $(\hat{\boldsymbol{q}}, a)$ is computable in polynomial time for every $\mathcal{O}$ in $\mathcal{L}$, we say that $\mathcal{L}$ *admits tractable containment reduction*. In the next theorem, we illustrate this definition by a few examples.

Lemma 1. (1) *FO without equality admits tractable containment reduction*; *in particular, both DL-Lite$_\mathcal{H}$ and $\mathcal{ALCHI}$ admits tractable containment reduction.*

(2) *DL-Lite$_\mathcal{F}$ admits tractable containment reduction.*

(3) *The ontology $\mathcal{O}$ with the single axiom*

$$\forall x, y_1, y_2, y_3 \,\big(\bigwedge_{i=1,2,3} P(x, y_i) \to \bigvee_{i \neq j} (y_i = y_j) \big)$$

(*or $\mathcal{O} = \{\geq 3\, P \sqsubseteq \bot\}$ in the DL parlance*) *does not admit containment reduction.*

Proof. (1) We use the following fact [15, Proposition 5.9]: for any FO-ontology $\mathcal{O}$ without =, any CQ $\boldsymbol{q}$, and any pointed data instances $\mathcal{A}_1, a_1$ and $\mathcal{A}_2, a_2$, if there is a homomorphism $h\colon \mathcal{A}_1 \to \mathcal{A}_2$ with $h(a_1) = a_2$, then $\mathcal{O}, \mathcal{A}_1 \models \boldsymbol{q}(a_1)$ implies $\mathcal{O}, \mathcal{A}_2 \models \boldsymbol{q}(a_2)$.

Let $(\hat{\boldsymbol{q}}(x), a)$ be *induced* by $\boldsymbol{q}(x)$, that is, obtained from $\boldsymbol{q}$ by replacing its variables by distinct constants, with x replaced by a. Suppose $\mathcal{O}, \hat{\boldsymbol{q}} \models \boldsymbol{q}'$ and $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}(a)$ but $\mathcal{O}, \mathcal{A} \not\models \boldsymbol{q}'(a)$. Take a model $\mathcal{I}$ witnessing $\mathcal{O}, \mathcal{A} \not\models \boldsymbol{q}_2(a)$. Then $\mathcal{I} \models \boldsymbol{q}_1(a)$ and this is witnessed by a homomorphism $h : \boldsymbol{q}_1 \to \mathcal{I}$. Take the image $h(\boldsymbol{q}_1)$. Then $\mathcal{O}, \widehat{h(\boldsymbol{q}_1)} \not\models \boldsymbol{q}_2(x)$ is witnessed by $\mathcal{I}$, and so $\mathcal{O}, \hat{\boldsymbol{q}}_1 \not\models \boldsymbol{q}_2(x)$, which is a contradiction.

(2) Let $\mathcal{O}$ be a *DL-Lite$_\mathcal{F}$* ontology. Given a CQ $\boldsymbol{q}(x)$, define an equivalence relation $\sim$ on *var*$(\boldsymbol{q})$ as the transitive closure of the following relation: $y \sim' z$ iff there is $u \in$ *var*$(\boldsymbol{q})$ such that $S(u, y), S(u, z) \in \boldsymbol{q}$, for a functional S in $\mathcal{O}$. Let $\boldsymbol{q}/_\sim$ be obtained by identifying (glueing together) all of the variables in each equivalence class $y/_\sim$. Clearly, $\boldsymbol{q}/_\sim(x/_\sim)$ is a homomorphic image of $\boldsymbol{q}(x)$ and $\boldsymbol{q}(x) \equiv_\mathcal{O} \boldsymbol{q}/_\sim(x/_\sim)$. We define $(\hat{\boldsymbol{q}}, a)$ as the pointed data instance induced by $\boldsymbol{q}/_\sim(x/_\sim)$. Conditions (cr$_1$) and (cr$_2$) are obvious, and (cr$_3$) follows from the fact that $\boldsymbol{q}/_\sim \models_\mathcal{O} \boldsymbol{q}'/_\sim$ iff $\boldsymbol{q}/_\sim \models_{\mathcal{O}'} \boldsymbol{q}'/_\sim$, where $\mathcal{O}'$ is obtained from $\mathcal{O}$ by omitting all of its functionality constraints, which is in the scope of part (1).

(3) Consider the CQ $\boldsymbol{q}(x) = \{P(x, y_i), A_i(y_i) \mid i = 1, 2, 3\}$ and suppose there is a suitable pointed data instance $(\hat{\boldsymbol{q}}, a)$. As $\mathcal{O}$ and the data instance induced by $\boldsymbol{q}$ are not satisfiable and in view of (cr$_2$), $\hat{\boldsymbol{q}}$ contains at most three individuals, say, $\hat{\boldsymbol{q}} = \{P(a, b), A_1(b), A_2(b), P(a, c), A_3(c)\}$. But then, by (cr$_3$), $\boldsymbol{q}'(x) = \{P(x, y), A_1(y), A_2(y), P(x, z), A_3(z)\}$ should satisfy $\boldsymbol{q} \models_\mathcal{O} \boldsymbol{q}'$, which not the case as witnessed by $\mathcal{A} = \{P(a, b), A_1(b), P(a, c), A_2(c), A_3(c)\}$ because $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}(a)$ but $\mathcal{O}, \mathcal{A} \not\models \boldsymbol{q}'(a)$. ☐

In what follows, we only consider ontology languages that admit containment reduction and assume that $(\hat{\boldsymbol{q}}, a)$ is a pointed data instance satisfying (**cr**$_1$)–(**cr**$_3$) for given $\boldsymbol{q}$ and $\mathcal{O}$.

An *example set* is a pair $E = (E^+, E^-)$, where E^+ and E^- are finite sets of pointed data instances $(\mathcal{D}, a)$. A CQ $\boldsymbol{q}(x)$ *fits* E wrt $\mathcal{O}$ if $\mathcal{O}, \mathcal{D}^+ \models \boldsymbol{q}(a^+)$ and $\mathcal{O}, \mathcal{D}^- \not\models \boldsymbol{q}(a^-)$, for all $(\mathcal{D}^+, a^+) \in E^+$ and $(\mathcal{D}^-, a^-) \in E^-$. We say that E *uniquely characterises* $\boldsymbol{q}$ *wrt* $\mathcal{O}$ *within a given class* $\mathcal{Q}$ *of queries* if $\boldsymbol{q}$ fits E and $\boldsymbol{q} \equiv_\mathcal{O} \boldsymbol{q}'$, for any $\boldsymbol{q}' \in \mathcal{Q}$ fitting E. Note that if $E^+ = \emptyset$, then $\boldsymbol{q} \equiv_\mathcal{O} \bot$, and so $\boldsymbol{q}$ is not satisfiable wrt $\mathcal{O}$. We call a unique characterisation $E = (E^+, E^-)$ of $\boldsymbol{q}$ wrt $\mathcal{O}$ *singular*$^+$ if $E^+ = \{(\hat{\boldsymbol{q}}, a)\}$.

Lemma 2. *Suppose $\mathcal{O}$ admits containment reduction, $\boldsymbol{q} \in \mathcal{Q}$ is satisfiable wrt $\mathcal{O}$ and has a unique characterisation $E = (E^+, E^-)$ wrt $\mathcal{O}$ within $\mathcal{Q}$. Then $E' = ((\{\hat{\boldsymbol{q}}, a)\}, E^-)$ is a unique characterisation of $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$, too. Thus, every uniquely characterisable $\boldsymbol{q} \in \mathcal{Q}$ within $\mathcal{Q}$ wrt an ontology admitting containment reduction has a singular$^+$characterisation.*

Proof. To show that E' is as required, we first observe that $\boldsymbol{q}$ fits E' by (**cr**$_1$) and (**cr**$_3$). Suppose $\boldsymbol{q}' \not\equiv_\mathcal{O} \boldsymbol{q}$ for some $\boldsymbol{q}' \in \mathcal{Q}$. We show that then either $\mathcal{O}, \hat{\boldsymbol{q}} \not\models \boldsymbol{q}'$ or $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}'$ for some $\mathcal{D} \in E^-$. Let $\boldsymbol{q} \not\models_\mathcal{O} \boldsymbol{q}'$. Then $\mathcal{O}, \hat{\boldsymbol{q}} \not\models \boldsymbol{q}'$ by (**cr**$_3$). Let $\boldsymbol{q} \models_\mathcal{O} \boldsymbol{q}'$ and $\boldsymbol{q}' \not\models_\mathcal{O} \boldsymbol{q}$. Then $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}'$ for all $\mathcal{D} \in E^+$, and so $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}'$, for some $\mathcal{D} \in E^-$, because E is a unique characterisation of $\boldsymbol{q}$ wrt $\mathcal{O}$. ☐

We now give two sufficient conditions of singular$^+$characterisability: the existence of frontiers and the existence of split-partners.

Let $\mathcal{O}$ be an ontology, $\mathcal{Q}$ a class of queries, and $\boldsymbol{q} \in \mathcal{Q}$ a satisfiable query wrt $\mathcal{O}$. A *frontier of $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$* is any set $\mathcal{F}_{\boldsymbol{q}} \subseteq \mathcal{Q}$ such that

- $\boldsymbol{q} \models_\mathcal{O} \boldsymbol{q}'$ and $\boldsymbol{q}' \not\models_\mathcal{O} \boldsymbol{q}$, for all $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$;
- if $\boldsymbol{q} \models_\mathcal{O} \boldsymbol{q}''$ for a $\boldsymbol{q}'' \in \mathcal{Q}$, then $\boldsymbol{q}'' \models_\mathcal{O} \boldsymbol{q}$ or there is $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$ with $\boldsymbol{q}' \models_\mathcal{O} \boldsymbol{q}''$.

(Note that if $\boldsymbol{q} \equiv_{\mathcal{O}} \top$, then $\mathcal{F}_{\boldsymbol{q}} = \emptyset$.) An ontology $\mathcal{O}$ is said to *admit* (*finite*) *frontiers within* $\mathcal{Q}$ if every $\boldsymbol{q} \in \mathcal{Q}$ satisfiable wrt $\mathcal{O}$ has a (finite) frontier wrt $\mathcal{O}$ within $\mathcal{Q}$. Further, if such frontiers can be computed in polynomial time, we say that $\mathcal{O}$ *admits polynomial-time computable frontiers.*

Theorem 8. *Suppose $\mathcal{Q}$ is a class of queries, an ontology $\mathcal{O}$ admits containment reduction, $\boldsymbol{q} \in \mathcal{Q}$ is satisfiable wrt $\mathcal{O}$, and $\mathcal{F}_{\boldsymbol{q}}$ is a finite frontier of $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$. Then $(\{(\hat{\boldsymbol{q}}, a)\}, \{(\hat{\boldsymbol{r}}, a) \mid \boldsymbol{r} \in \mathcal{F}_{\boldsymbol{q}}\})$ is a singular$^+$characterisation of $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$.*

Proof. By **(cr$_2$)**, $\mathcal{O}, \hat{\boldsymbol{q}} \models \boldsymbol{q}(a)$. To show $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{q}(a)$ for all $\boldsymbol{r} \in \mathcal{F}_{\boldsymbol{q}}$, we observe that $\boldsymbol{r} \not\models_{\mathcal{O}} \boldsymbol{q}$ by the definition of $\mathcal{F}_{\boldsymbol{q}}$, so $\boldsymbol{r}(x)$ is consistent with $\mathcal{O}$ and by **(cr$_3$)** for $\boldsymbol{r}$, from which $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{q}(a)$. Thus, $\boldsymbol{q}$ fits E.

Suppose $\boldsymbol{q}' \in \mathcal{Q}$ and $\boldsymbol{q} \not\equiv_{\mathcal{O}} \boldsymbol{q}'$. We show that either $\mathcal{O}, \hat{\boldsymbol{q}} \not\models \boldsymbol{q}'$ or $\mathcal{O}, \hat{\boldsymbol{r}} \models \boldsymbol{q}'$ for some $\boldsymbol{r} \in \mathcal{F}_{\boldsymbol{q}}$. If $\boldsymbol{q} \not\models_{\mathcal{O}} \boldsymbol{q}'$, then, since $\mathcal{O}$ admits containment reduction and $\boldsymbol{q}(x)$ is satisfiable wrt $\mathcal{O}$, we obtain $\mathcal{O}, \hat{\boldsymbol{q}} \not\models \boldsymbol{q}'$ by **(cr$_3$)**. So suppose $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}'$ and $\boldsymbol{q}' \not\models_{\mathcal{O}} \boldsymbol{q}$. As $\mathcal{F}_{\boldsymbol{q}}$ is a frontier of $\boldsymbol{q}$ wrt $\mathcal{O}$, there is $\boldsymbol{r} \in \mathcal{F}_{\boldsymbol{q}}$ with $\boldsymbol{r} \models_{\mathcal{O}} \boldsymbol{q}'$. If $\boldsymbol{r}(x)$ is unsatisfiable wrt $\mathcal{O}$, then $\mathcal{O}$ and $\hat{\boldsymbol{r}}$ are unsatisfiable by **(cr$_1$)**, and so $\mathcal{O}, \hat{\boldsymbol{r}} \models \boldsymbol{q}'$. And if $\boldsymbol{r}(x)$ is satisfiable wrt $\mathcal{O}$, we obtain $\mathcal{O}, \hat{\boldsymbol{r}} \models \boldsymbol{q}'(a)$ by **(cr$_3$)**. ☐

The two main ontology languages that admit polynomial-time computable frontiers within ELIQ are *DL-Lite*$_{\mathcal{H}}$ and *DL-Lite*$^-_{\mathcal{F}}$, whereas *DL-Lite*$_{\mathcal{F}}$ itself does not admit finite ELIQ-frontiers [26]. By Theorem 8 and Lemma 1, we then obtain:

Theorem 9 ([26])**.** *If an ELIQ $\boldsymbol{q}$ is satisfiable wrt a DL-Lite$_{\mathcal{H}}$ or DL-Lite$^-_{\mathcal{F}}$ ontology $\mathcal{O}$, then $\boldsymbol{q}$ has a polysize singular$^+$characterisation wrt $\mathcal{O}$ within ELIQ.*

We also require the following observation on singleton frontiers. A query $\boldsymbol{r} \in \mathcal{Q}$ is called *meet-reducible* [38] (or *conjunct*) wrt $\mathcal{O}$ within $\mathcal{Q}$ if there are $\boldsymbol{r}_1, \boldsymbol{r}_2 \in \mathcal{Q}$ such that

$$\boldsymbol{r} \equiv_{\mathcal{O}} \boldsymbol{r}_1 \wedge \boldsymbol{r}_2 \quad \text{and} \quad \boldsymbol{r} \not\equiv_{\mathcal{O}} \boldsymbol{r}_i, \quad \text{for } i = 1, 2.$$

Lemma 3. (i) *If an ontology $\mathcal{O}$ admits frontiers within $\mathcal{Q}$, then $\boldsymbol{q} \in \mathcal{Q}$ is meet-reducible wrt $\mathcal{O}$ within $\mathcal{Q}$ iff $|\mathcal{F}_{\boldsymbol{q}}| \geq 2$ provided that, for any distinct $\boldsymbol{q}', \boldsymbol{q}'' \in \mathcal{F}_{\boldsymbol{q}}$, we have $\boldsymbol{q}' \not\models_{\mathcal{O}} \boldsymbol{q}''$.*

(ii) *If an ontology $\mathcal{O}$ admits singular$^+$characterisations within $\mathcal{Q}$, then, for any meet-reducible $\boldsymbol{q} \in \mathcal{Q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$, we have $|\mathcal{N}_{\boldsymbol{q}}| \geq 2$ in every singular$^+$ characterisation $(\{\hat{\boldsymbol{q}}\}, \mathcal{N}_{\boldsymbol{q}})$ of $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$.*

Proof. $(i, \Leftarrow)$ Let $\boldsymbol{r}_1, \boldsymbol{r}_2 \in \mathcal{F}_{\boldsymbol{q}}$ be distinct. Consider $\boldsymbol{r} = \boldsymbol{r}_1 \wedge \boldsymbol{r}_2$. If $\boldsymbol{q} \not\equiv_{\mathcal{O}} \boldsymbol{r}$, then there is $\boldsymbol{r}' \in \mathcal{F}_{\boldsymbol{q}}$ with $\boldsymbol{r}' \models_{\mathcal{O}} \boldsymbol{r}$, and so $\boldsymbol{r}' \models_{\mathcal{O}} \boldsymbol{r}_i$, which is impossible.

$(i, \Rightarrow)$ Suppose $\boldsymbol{q} \equiv_{\mathcal{O}} \boldsymbol{q}_1 \wedge \boldsymbol{q}_2$ and $\boldsymbol{q} \not\equiv_{\mathcal{O}} \boldsymbol{q}_i$, for $i = 1, 2$. Then there are $\boldsymbol{r}_i \in \mathcal{F}_{\boldsymbol{q}}$ with $\boldsymbol{r}_i \not\equiv_{\mathcal{O}} \boldsymbol{q}_i$. Clearly, $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are distinct because otherwise $\boldsymbol{r}_i \models_{\mathcal{O}} \boldsymbol{q}$.

(ii) Suppose $\boldsymbol{q} \equiv_{\mathcal{O}} \boldsymbol{q}_1 \wedge \boldsymbol{q}_2$ and $\boldsymbol{q} \not\equiv_{\mathcal{O}} \boldsymbol{q}_i$, for $i = 1, 2$. If $\mathcal{N}_{\boldsymbol{q}} = \{\hat{\boldsymbol{r}}\}$, then $\mathcal{O}, \hat{\boldsymbol{r}} \models \boldsymbol{q}_i$, for $i = 1, 2$, because if $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{q}_i$, then $\boldsymbol{q}_i$ would fit $(\{\hat{\boldsymbol{q}}\}, \mathcal{N}_{\boldsymbol{q}})$, and so equivalent to $\boldsymbol{q}$ wrt $\mathcal{O}$, which is impossible. ☐

Lemma 4. (i) *Deciding whether an ELIQ $\boldsymbol{q}$ is meet-reducible wrt to a DL-Lite$^-_{\mathcal{F}}$-ontology can be done in polynomial time.*

(ii) *Deciding whether an ELIQ $\boldsymbol{q}$ is meet-reducible wrt to a DL-Lite$_{\mathcal{H}}$-ontology is* NP*-complete.*

Proof. (i) We first compute a frontier $\mathcal{F}_{\boldsymbol{q}}$ of $\boldsymbol{q}$ in polynomial time [26]. Then we remove from $\mathcal{F}_{\boldsymbol{q}}$ every $\boldsymbol{q}''$ for which there is a different $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$ with $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}''$. This can also be done in polynomial time because query containment in *DL-Lite*$_{\mathcal{F}}$ is tractable. It remains to use Lemma 3 (i) to check if the resulting set is a singleton.

(ii) Consider the ontologies $\mathcal{O}$, ABox $\{A_0(a)\}$, and $\mathcal{ELI}$-queries $\boldsymbol{q}$ constructed in the proof of [32, Theorem 1] for Boolean CNFs. As shown in that proof, the problem $\mathcal{O}, A_0(a) \models \boldsymbol{q}(a)$ is NP-hard. If we take copies $\mathcal{O}'$ and $\boldsymbol{q}'$ by replacing all predicates except A_0 by fresh predicates, then $\boldsymbol{q} \wedge \boldsymbol{q}'$ is a lone conjunct iff $\mathcal{O} \cup \mathcal{O}', A_0(a) \not\models \boldsymbol{q}(a)$. ☐

We next introduce splittings (aka homomorphism dualities). Suppose $\mathcal{Q}^\sigma$ is a class of σ-queries, for a finite signature σ of predicate symbols, $\mathcal{O}$ an ontology in the signature σ, and $Q \subseteq \mathcal{Q}^\sigma$ is a finite set queries. A set $\mathcal{S}(Q)$ of pointed σ-data instances $(\mathcal{A}, a)$ is called a *split-partner* for Q wrt $\mathcal{O}$ within $\mathcal{Q}^\sigma$ if, for all $\boldsymbol{q}' \in \mathcal{Q}^\sigma$, we have

$$\mathcal{O}, \mathcal{A} \models \boldsymbol{q}'(a) \text{ for some } (\mathcal{A}, a) \in \mathcal{S}(Q) \quad \text{iff} \quad \boldsymbol{q}' \not\models_{\mathcal{O}} \boldsymbol{q} \text{ for all } \boldsymbol{q} \in Q. \tag{3}$$

An ontology language $\mathcal{L}$ is said to *have general split-partners within* $\mathcal{Q}^\sigma$ if all finite sets of $\mathcal{Q}^\sigma$-queries have split partners wrt any $\mathcal{L}$-ontology in the signature σ. If this holds for all singleton subsets of $\mathcal{Q}^\sigma$, we say that $\mathcal{L}$ *has split-partners within* $\mathcal{Q}^\sigma$.

We illustrate the notion of split-partner by a few examples, the last of which shows that without the restriction to a finite signature σ, split-partners almost never exist.

Example 1. (i) Let $\mathcal{O}$ be any ontology such that $\mathcal{A}$ and $\mathcal{O}$ are satisfiable for all data instances $\mathcal{A}$; for example, $\mathcal{O} = \{A \sqsubseteq B\}$. Let $\mathcal{Q}^\sigma$ be any class of σ-CQs, for some signature σ. Then the split-partner $\mathcal{S}_\perp$ of $\perp(x)$ wrt $\mathcal{O}$ in $\mathcal{Q}^\sigma$ is the set

$$\mathcal{S}_\perp = \{\mathcal{B}_\sigma\} \quad \text{with} \quad \mathcal{B}_\sigma = \{R(a,a) \mid R \in \sigma\} \cup \{A(a) \mid A \in \sigma\}.$$

Clearly, $\mathcal{O}, \mathcal{B}_\sigma \models \boldsymbol{q}$, for any $\boldsymbol{q} \in \mathcal{Q}^\sigma$ different from $\perp$.

(ii) For $\mathcal{O} = \{A \sqcap B \sqsubseteq \perp\}$ and $\sigma = \{A, B\}$, we have $\mathcal{S}_\perp = \{\{A(a)\},\ \{B(a)\}\}$.

(iii) There does not exist a split-partner for $Q = \{A(x)\}$ wrt the empty ontology $\mathcal{O}$ within ELIQ. To show this, observe that $B(x) \not\models_\mathcal{O} A(x)$ for any unary predicate $B \neq A$. Hence, as any data instance $\mathcal{A}$ is finite, there is no finite set $\mathcal{S}(\{A(x)\})$ satisfying (3).

In contrast, for frontiers and unique characterisations, restrictions to sets of predicate symbols containing all symbols in the query and ontology do not make any difference. Indeed, let σ contain all symbols in $\mathcal{O}$ and $\boldsymbol{q}$. Then, for any class $\mathcal{Q}$ of queries, a set $\mathcal{F}_{\boldsymbol{q}}$ is a frontier for $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$ if it is a frontier for $\boldsymbol{q}$ wrt $\mathcal{O}$ within the restriction of $\mathcal{Q}$ to σ. The reason is that if $\boldsymbol{q} \models_\mathcal{O} \boldsymbol{q}'$ for a CQ $\boldsymbol{q}'$ then $\boldsymbol{q}'$ does not contain any symbols not in $\boldsymbol{q}$ or $\mathcal{O}$ if $\boldsymbol{q}$ is satisfiable wrt $\mathcal{O}$. The same holds for characterisations E of $\boldsymbol{q}$ wrt $\mathcal{O}$.

Theorem 10. *$\mathcal{ALCHI}$ has general split-partners within ELIQ$^\sigma$, which can be computed in exponential time.*

Proof. Suppose a finite set $Q \subseteq$ ELIQ$^\sigma$ and an $\mathcal{ALCHI}$-ontology $\mathcal{O}$ in the signature σ are given. Let $sub_{\mathcal{O},Q}$ be the closure under single negation of the set of subconcepts of concepts in Q and $\mathcal{O}$. A *type for* $\mathcal{O}$ is any maximal subset $tp \subseteq sub_{\mathcal{O},Q}$ consistent with $\mathcal{O}$. Let $\boldsymbol{T}$ be the set of all types for $\mathcal{O}$. Define a σ-data instance $\mathcal{A}$ with $ind(\mathcal{A}) = \boldsymbol{T}$, $A(tp) \in \mathcal{A}$ for all concept names $A \in \sigma$ and tp such that $A \in tp$, and $P(tp, tp') \in \mathcal{A}$ for all role names $P \in \sigma$, tp and tp' such that tp and tp' can be satisfied by the domain elements of a model of $\mathcal{O}$ that are related via P. We consider an interpretation $\mathcal{I}_\mathcal{A}$ with $\Delta^{\mathcal{I}_\mathcal{A}} = \{ind(\mathcal{A})\}$, $A^{\mathcal{I}_\mathcal{A}} = \{tp \mid A(tp) \in \mathcal{A}\}$ for concept names $A \in \sigma$, $A^{\mathcal{I}_\mathcal{A}} = \emptyset$ for $A \notin \sigma$, $P^{\mathcal{I}_\mathcal{A}} = \{(tp, tp') \mid P(tp, tp') \in \mathcal{A}\}$ for role names $P \in \sigma$, $P^{\mathcal{I}_\mathcal{A}} = \emptyset$ for $P \notin \sigma$. It can be readily checked that, for any $\boldsymbol{q} \in \mathcal{Q}^\sigma$,

$$\mathcal{I}_\mathcal{A} \models \mathcal{O}, \mathcal{A}, \tag{4}$$

$$\boldsymbol{q} \in tp \quad \text{iff} \quad \mathcal{I}_\mathcal{A}, tp \models \boldsymbol{q} \quad \text{iff} \quad \mathcal{O}, \mathcal{A} \models \boldsymbol{q}(tp). \tag{5}$$

Let $Q = \{\boldsymbol{q}_1, \dots, \boldsymbol{q}_n\}$ and let $\mathcal{A}^n$ be the n-times direct product of $\mathcal{A}$. Set

$$\mathcal{S}(Q) = \{(\mathcal{A}^n, (tp_1, \dots, tp_n)) \mid \neg\boldsymbol{q}_i \in tp_i \text{ for all } 1 \leq i \leq n\}.$$

We prove (3) for an arbitrary $\boldsymbol{q}' \in \mathcal{Q}^\sigma$. For the $(\Rightarrow)$ direction, suppose $\mathcal{O}, \mathcal{A}^n \models \boldsymbol{q}'(\vec{tp})$ for some $(\mathcal{A}^n, \vec{tp}) \in \mathcal{S}(Q)$ and $\vec{tp} = (tp_1, \dots, tp_n)$. Fix $i \in \{1, \dots, n\}$. Observe that the *projection* map $h((tp'_1, \dots, tp'_n)) = tp'_i$ for $(tp'_1, \dots, tp'_n) \in \boldsymbol{T}^n$ is a homomorphism from $\mathcal{A}^n$ to $\mathcal{A}$ such that $h(\vec{tp}) = tp_i$. As in the proof of Lemma 1 (1), we obtain $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}'(tp_i)$. Recall that $\neg\boldsymbol{q}_i \in tp_i$. Then, by (5), we have $\mathcal{I}_\mathcal{A}, tp_i \models \boldsymbol{q}'$, $\mathcal{I}_\mathcal{A}, tp_i \not\models \boldsymbol{q}_i$, and so using (4) we obtain $\boldsymbol{q}' \not\models_\mathcal{O} \boldsymbol{q}_i$. For the opposite direction, suppose $\boldsymbol{q}' \not\models_\mathcal{O} \boldsymbol{q}_i$ for all $1 \leq i \leq n$. It follows that, for each i, there exists $tp_i \in \boldsymbol{T}$ such that $\boldsymbol{q}', \neg\boldsymbol{q}_i \in tp_i$. Let $\vec{tp} = (tp_1, \dots, tp_n)$. Clearly, $(\mathcal{A}^n, \vec{tp}) \in \mathcal{S}(Q)$ and it remains to show that $\mathcal{O}, \mathcal{A}^n \models \boldsymbol{q}'(\vec{tp})$. Observe that, for each tp_i, by (5), there exists a homomorphism h_i that maps $\boldsymbol{q}'$ into $\mathcal{I}_\mathcal{A}$ with its root mapped to tp_i. By the construction of $\mathcal{I}_\mathcal{A}$, the same holds for $\mathcal{A}$ in place of $\mathcal{I}_\mathcal{A}$. Because $\mathcal{A}^n$ is a direct product, there exists a homomorphism that maps $\boldsymbol{q}'$ into $\mathcal{A}^n$ with its root mapped to $\vec{tp}$. Thus, $\mathcal{A}^n \models \boldsymbol{q}'(\vec{tp})$. □

Our second sufficient condition of unique characterisability is as follows:

Theorem 11. *Suppose $\mathcal{Q}$ is a class of queries, $\mathcal{O}$ an ontology that admits containment reduction, $\boldsymbol{q} \in \mathcal{Q}$ is satisfiable wrt $\mathcal{O}$, and σ is a signature containing the predicate symbols in $\boldsymbol{q}$ and $\mathcal{O}$. If $\mathcal{S}_{\boldsymbol{q}}$ is a split-partner for $\{\boldsymbol{q}\}$ wrt $\mathcal{O}$ within $\mathcal{Q}^\sigma$, then $(\{(\hat{\boldsymbol{q}}, a)\}, \mathcal{S}_{\boldsymbol{q}})$ is a singular$^+$characterisation of $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathcal{Q}$.*

Proof. Clearly, $\boldsymbol{q}$ fits E as $\mathcal{O}, \hat{\boldsymbol{q}} \models \boldsymbol{q}(a)$ and $\mathcal{O}, \mathcal{A} \not\models \boldsymbol{q}(a)$ for any $(\mathcal{A}, a) \in \mathcal{S}_{\boldsymbol{q}}$ as otherwise $\boldsymbol{q} \not\models_\mathcal{O} \boldsymbol{q}$. Now, suppose $\boldsymbol{q}' \not\equiv_\mathcal{O} \boldsymbol{q}$. If $\boldsymbol{q}' \models_\mathcal{O} \boldsymbol{q}$, then $\boldsymbol{q} \not\models_\mathcal{O} \boldsymbol{q}'$, and so $\mathcal{O}, \hat{\boldsymbol{q}} \not\models \boldsymbol{q}'(a)$. Hence $\boldsymbol{q}'$ does not fit E. If $\boldsymbol{q}' \not\models_\mathcal{O} \boldsymbol{q}$, then there exists $(\mathcal{A}, a) \in \mathcal{S}_{\boldsymbol{q}}$ with $\mathcal{O}, \mathcal{A} \models \boldsymbol{q}'(a)$, and so again $\boldsymbol{q}'$ does not fit E. □

As a consequence of Theorem 10, 11 and Lemma 1, we obtain the following:

Theorem 12. *If $\boldsymbol{q} \in$ ELIQ$^\sigma$ is satisfiable wrt an $\mathcal{ALCHI}$-ontology $\mathcal{O}$ in a signature σ, then $\boldsymbol{q}$ has a singular$^+$characterisation wrt $\mathcal{O}$ within ELIQ$^\sigma$.*

The sufficient conditions for singular$^+$characterisations of Theorems 8 and 11 use the notions of frontier and split-partner, respectively. We now give examples showing that there are rather simple queries (and ontologies) with frontiers but no split-partners and vice versa. Hence these two notions are not equivalent. The query witnessing that frontiers can exist where split-partners do not exist provides a counterexample even if one admits frontiers containing CQs. In more detail, a *CQ-frontier* for an ELIQ $\boldsymbol{q}$ wrt to an ontology $\mathcal{O}$ is a set $\mathcal{F}_{\boldsymbol{q}}$ of CQs such that

- if $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}''$, for a CQ $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$ and an ELIQ $\boldsymbol{q}''$, then $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}''$ and $\boldsymbol{q}'' \not\models_{\mathcal{O}} \boldsymbol{q}$;
- if $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}''$ and $\boldsymbol{q}'' \not\models_{\mathcal{O}} \boldsymbol{q}$, for an ELIQ $\boldsymbol{q}''$, then there exists $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$ such that $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}''$.

Clearly standard ELIQ frontiers defined above are also CQ-frontiers.

Theorem 13. *$\mathcal{EL}$ does not admit finite CQ-frontiers within ELIQ.*

Proof. We show that the query $\boldsymbol{q} = A \wedge B$ does not have a finite CQ-frontier wrt the ontology

$$\mathcal{O} = \{A \sqsubseteq \exists R.A,\ \exists R.A \sqsubseteq A\}$$

within ELIQs. Suppose otherwise. Let $\mathcal{F}_{\boldsymbol{q}}$ be such a CQ-frontier. Consider the ELIQs $\boldsymbol{r}_{n,m} = \exists R^n \exists R^{-m}.B$ with $n > m > 0$. Clearly, $\boldsymbol{q} \not\models_{\mathcal{O}} \boldsymbol{r}_{n,m}$, and so $\boldsymbol{r}_{n,m}$ cannot be entailed wrt $\mathcal{O}$ by any CQ in $\mathcal{F}_{\boldsymbol{q}}$. Thus, if $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$ and $B(x)$ is in $\boldsymbol{q}'(x)$, then we cannot have an R-cycle in $\boldsymbol{q}'$ reachable from x along an R-path as otherwise we would have $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{r}_{n,m}$ for suitable n, m.

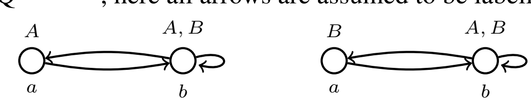
Consider now the ELIQs $\boldsymbol{q}_n = B \wedge \exists R^n.\top$, for all $n \geq 1$. Clearly, $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}_n$. As $\boldsymbol{q}_n \not\models_{\mathcal{O}} \boldsymbol{q}$, infinitely many $\boldsymbol{q}_n$ are entailed by some $\boldsymbol{q}' \in \mathcal{F}_{\boldsymbol{q}}$ wrt $\mathcal{O}$. Take such a $\boldsymbol{q}'$. Since $B(x) \in \boldsymbol{q}'$ because of $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}_n$, no R-cycle is reachable from x via an R-path in $\boldsymbol{q}'$. Note also that no y with $A(y) \in \boldsymbol{q}'$ can be reached from x along an R-path as otherwise $\boldsymbol{q}' \models_{\mathcal{O}} B \wedge \exists R^k.A$ for some $k \geq 0$ and, since $B \wedge \exists R^k.A \models_{\mathcal{O}} \boldsymbol{q}$ by the second axiom in $\mathcal{O}$, we would have $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}$.

To derive a contradiction, we show now that there is an R-path of any length n starting at x in $\boldsymbol{q}'$. Suppose this is not the case. Let n be the length of a longest R-path starting at x in $\boldsymbol{q}'$. We construct a model $\mathcal{I}$ of $\mathcal{O}$ and $\boldsymbol{q}'$ refuting $\boldsymbol{q}_{n+l}$, for any $l \geq 1$. Define $\mathcal{I}$ by taking

- $\Delta^{\mathcal{I}} = \mathit{var}(\boldsymbol{q}') \cup \{d_1, d_2, \dots\}$, for fresh d_i;
- $a \in B^{\mathcal{I}}$ if $B(a) \in \boldsymbol{q}'$;
- $a \in A^{\mathcal{I}}$ if there is an R-path in $\boldsymbol{q}'$ from a to some y with $A(y) \in \boldsymbol{q}'$ or $a = d_i$ for some i;
- $(a, b) \in R^{\mathcal{I}}$ if $R(a, b) \in \boldsymbol{q}'$ or there is an R-path in $\boldsymbol{q}'$ from a to some y with $A(y) \in \boldsymbol{q}'$ and $b = d_1$, or $a = d_i$ and $b = d_{i+1}$.

By the construction and the fact that no y with $A(y) \in \boldsymbol{q}'$ can be reached from x along an r-path, $\mathcal{I}$ is a model of $\mathcal{O}$ and $\boldsymbol{q}'$ refuting $\boldsymbol{q}_{n+l}$. □

Example 2. On the other hand, the following set of pointed data instances is a split-partner of $\{\boldsymbol{q}\}$ wrt $\mathcal{O}$ from the proof of Theorem 13 within ELIQ$^{\{A,B,R\}}$; here all arrows are assumed to be labelled with R:



Theorem 14. *There exist a DL-Lite$^-_{\mathcal{F}}$ ontology $\mathcal{O}$, a query $\boldsymbol{q}$ and a signature σ such that $\{\boldsymbol{q}\}$ does not have a finite split-partner wrt $\mathcal{O}$ within ELIQ$^\sigma$.*

Proof. Consider $\boldsymbol{q} = A$ and $\mathcal{O} = \{\mathrm{fun}(P), \mathrm{fun}(P^-), B \sqcap \exists P^- \sqsubseteq \bot\}$. We show that $Q = \{\boldsymbol{q}\}$ does not have a finite split partner wrt $\mathcal{O}$ within ELIQ$^{\{A,B,P\}}$. Suppose otherwise and $\mathcal{S}(Q)$ is such a split-partner. Then there is $(\mathcal{A}, a) \in \mathcal{S}(Q)$ with $\mathcal{O}, \mathcal{A} \models B \sqcap \exists P^n.\top(a)$ for all sufficiently large n because $B \sqcap \exists P^n.\top \not\models_{\mathcal{O}} A$. Then $\mathcal{A}$ must contain n nodes if $\mathcal{O}$ and $\mathcal{A}$ are satisfiable, so $\mathcal{S}(Q)$ is infinite.

On the other hand, $\{\top\}$ is a frontier for A wrt $\mathcal{O}$ within ELIQ. □

Our final example in this section combines the examples of the two previous theorems and shows that even by taking frontiers and splittings together we do not obtain a universal method for constructing singular$^+$unique characterisations.

Example 3. Consider the ELIQ $\boldsymbol{q} = A \wedge B$ and the $\mathcal{ELFI}^\bot$-ontology

$$\mathcal{O} = \{A \sqsubseteq \exists R.A,\ \exists R.A \sqsubseteq A,\ \text{fun}(P),\ \text{fun}(P^-),\ E \sqcap \exists P^- \sqsubseteq \bot\}.$$

It can be shown in the same way as above that $\boldsymbol{q}$ has no frontier wrt $\mathcal{O}$ within ELIQ and that $\boldsymbol{q}$ does not have any split-partner wrt $\mathcal{O}$ within ELIQ$^{\{A,B,R,P,E\}}$. However, a singular$^+$unique characterisation of $\boldsymbol{q}$ wrt $\mathcal{O}$ within ELIQ is obtained by taking $E^+ = \{\hat{\boldsymbol{q}}\}$ and E^- the same as in Example 2. (To show the latter one only has to observe that E^- is a split-partner of $\boldsymbol{q}$ wrt $\mathcal{O}$ within ELIQ$^{\{A,B,R\}}$ and that $\mathcal{O}, \hat{\boldsymbol{q}} \not\models \boldsymbol{r}$ for any $\boldsymbol{r}$ containing any of the symbols P or E.)

3.2 Two-dimensional Queries

A *temporal σ-data instance*, $\mathcal{D}$, is a finite sequence $\mathcal{A}_0, \dots, \mathcal{A}_n$ of σ-data instances, where, intuitively, each $\mathcal{A}_i$ comprises the facts with timestamp i. We assume that all *ind*$(\mathcal{A}_i)$ are the same, adding $\top(a)$ to $\mathcal{A}_i$ if needed, and set *ind*$(\mathcal{D})$ = *ind*$(\mathcal{A}_0)$. The *length* of $\mathcal{D}$ is $\max(\mathcal{D}) = n$ and the *size* of $\mathcal{D}$ is $|\mathcal{D}| = \sum_{i \leq n} |\mathcal{A}_i|$. Within a temporal σ-data instance, we often denote by $\emptyset$ the data instance $\{\top(a) \mid a \in \mathit{ind}(\mathcal{D})\}$.

Two-dimensional queries, $\boldsymbol{q}$, for accessing temporal data instances we propose in this paper are built from domain queries in a given class $\mathcal{Q}$ (say, ELIQs) using $\wedge$ and the (future-time) temporal operators from the standard linear temporal logic *LTL* over the time flow $(\mathbb{N}, <)$: unary $\bigcirc$ (at the next moment), $\Diamond$ (some time later), $\Diamond_r$ (now or later), and binary U (until); see below for the precise semantics. The class of such 2D-queries that only use the operators from a set $\Phi \subseteq \{\bigcirc, \Diamond, \Diamond_r, \mathsf{U}\}$ is denoted by $\mathit{LTL}^{\Phi}(\mathcal{Q})$. The class $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ comprises *path queries* of the form

$$\boldsymbol{q} = \boldsymbol{r}_0 \wedge \boldsymbol{o}_1(\boldsymbol{r}_1 \wedge \boldsymbol{o}_2(\boldsymbol{r}_2 \wedge \dots \wedge \boldsymbol{o}_n \boldsymbol{r}_n)), \tag{6}$$

where $\boldsymbol{o}_i \in \{\bigcirc, \Diamond, \Diamond_r\}$ and $\boldsymbol{r}_i \in \mathcal{Q}$; *path queries* in $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{Q})$ take the form

$$\boldsymbol{q} = \boldsymbol{r}_0 \wedge (\boldsymbol{l}_1 \,\mathsf{U}\, (\boldsymbol{r}_1 \wedge (\boldsymbol{l}_2 \,\mathsf{U}\, (\dots (\boldsymbol{l}_n \,\mathsf{U}\, \boldsymbol{r}_n) \dots)))), \tag{7}$$

where $\boldsymbol{r}_i \in \mathcal{Q}$ and either $\boldsymbol{l}_i \in \mathcal{Q}$ or $\boldsymbol{l}_i = \bot$. We use the character $\mathcal{C}$ to refer to classes of 2D queries. The *size* $|\boldsymbol{q}|$ of $\boldsymbol{q}$ is the number of symbols in $\boldsymbol{q}$, and the *temporal depth* *tdp*$(\boldsymbol{q})$ of $\boldsymbol{q}$ is the maximum number of nested temporal operators in $\boldsymbol{q}$.

An (atemporal) ontology $\mathcal{O}$ and temporal data instance $\mathcal{D} = \mathcal{A}_0, \dots, \mathcal{A}_n$ are *satisfiable* if $\mathcal{O}$ and $\mathcal{A}_i$ are satisfiable for each $i \leq n$. For satisfiable $\mathcal{O}$ and $\mathcal{D}$, the *entailment relation* $\mathcal{O}, \mathcal{D}, \ell, a \models \boldsymbol{q}$ with $\ell \in \mathbb{N}$ and $a \in \mathit{ind}(\mathcal{D})$ is defined by induction as follows, where $\mathcal{A}_\ell = \emptyset$, for $\ell > n$:

$$\begin{aligned}
&\mathcal{O}, \mathcal{D}, \ell, a \models \boldsymbol{q} \text{ iff } \mathcal{O}, \mathcal{A}_\ell \models \boldsymbol{q}(a), \text{ for any } \boldsymbol{q} \in \mathcal{Q},\\
&\mathcal{O}, \mathcal{D}, \ell, a \models \boldsymbol{q}_1 \wedge \boldsymbol{q}_2 \text{ iff } \mathcal{O}, \mathcal{D}, \ell, a \models \boldsymbol{q}_1 \text{ and } \mathcal{O}, \mathcal{D}, \ell, a \models \boldsymbol{q}_2,\\
&\mathcal{O}, \mathcal{D}, \ell, a \models \bigcirc\boldsymbol{q} \text{ iff } \mathcal{O}, \mathcal{D}, \ell+1, a \models \boldsymbol{q},\\
&\mathcal{O}, \mathcal{D}, \ell, a \models \Diamond\boldsymbol{q} \text{ iff } \mathcal{O}, \mathcal{D}, m, a \models \boldsymbol{q}, \text{ for some } m > \ell,\\
&\mathcal{O}, \mathcal{D}, \ell, a \models \Diamond_r\boldsymbol{q} \text{ iff } \mathcal{O}, \mathcal{D}, m, a \models \boldsymbol{q}, \text{ for some } m \geq \ell,\\
&\mathcal{O}, \mathcal{D}, \ell, a \models \boldsymbol{q}_1 \,\mathsf{U}\, \boldsymbol{q}_2 \text{ iff there is } m > \ell \text{ such that } \mathcal{O}, \mathcal{D}, m, a \models \boldsymbol{q}_2 \text{ and}\\
&\qquad\qquad \mathcal{O}, \mathcal{D}, k, a \models \boldsymbol{q}_1, \text{ for all } k, \ell < k < m.
\end{aligned}$$

If $\mathcal{O}$ and $\mathcal{D}$ are not satisfiable, we set $\mathcal{O}, \mathcal{D}, \ell, a \models \boldsymbol{q}$ for all $\boldsymbol{q}$, ℓ and a.

Observe that for ontologies $\mathcal{O}$, which are not Horn, there is an alternative natural semantics based on temporal structures. A *temporal structure* $\mathcal{I}$ is a sequence $\mathcal{I}_0, \mathcal{I}_1, \dots$ of domain structure $\mathcal{I}_i$ as introduced above such that $a^{\mathcal{I}_{n_1}} = a^{\mathcal{I}_{n_2}}$ for any individual a and $n_1, n_2 \in \mathbb{N}$. $\mathcal{I}$ is a model of $\mathcal{D} = \mathcal{A}_0, \dots, \mathcal{A}_n$ if each $\mathcal{I}_i$ is a model of $\mathcal{A}_i$ for $i \leq n$ and it is a model of $\mathcal{O}$ if each $\mathcal{I}_i$ is a model of $\mathcal{O}$. The truth relation $\mathcal{I}, \ell, a \models \boldsymbol{q}$ is defined in the obvious way and $\mathcal{O}, \mathcal{D}, \ell, a \models_c \boldsymbol{q}$ iff $\mathcal{I}, \ell, a \models \boldsymbol{q}$ for every model $\mathcal{I}$ of $\mathcal{O}$ and $\mathcal{D}$. It is easy to see that $\models$ coincides with $\models_c$ for any ontology $\mathcal{O}$ formulated in the Horn-fragment of FO [20], in particular all *DL-Lite* dialects considered in this paper and $\mathcal{ELHIF}^\bot$. Thus, the results presented in this paper also hold under $\models_c$ if one considers such ontologies. In general, however, the two entailment relations do not coincide: consider the ontology $\mathcal{O} = \{\top \sqsubseteq A \sqcup B\}$ and the data instance $\mathcal{D} = \emptyset, \mathcal{A}_1, \emptyset, \mathcal{A}_3$ with $\mathcal{A}_1 = \{A(a)\}$ and $\mathcal{A}_3 = \{B(a)\}$. Then $\mathcal{O}, \mathcal{D}, 0, a \models_c \Diamond(A \wedge \bigcirc B)$ but $\mathcal{O}, \mathcal{D}, 0, a \not\models \Diamond(A \wedge \bigcirc B)$. In what follows we consider $\models$ only and leave an investigation for $\models_c$ in the non-Horn case as future work.

An *example set* is a pair $E = (E^+, E^-)$ with finite sets E^+ and E^- of pointed temporal data instances $\mathcal{D}, a$ with $a \in \mathit{ind}(\mathcal{D})$. We say that a query $\boldsymbol{q}$ *fits* E wrt $\mathcal{O}$ if $\mathcal{O}, \mathcal{D}^+, 0, a^+ \models \boldsymbol{q}$ and $\mathcal{O}, \mathcal{D}^-, 0, a^- \not\models \boldsymbol{q}$, for all $\mathcal{D}^+, a^+ \in E^+$ and $\mathcal{D}^-, a^- \in E^-$. As before, E *uniquely characterises* $\boldsymbol{q}$ *wrt* $\mathcal{O}$ *within a class* $\mathcal{C}$ *of 2D-queries* if $\boldsymbol{q}$ fits E wrt $\mathcal{O}$ and every $\boldsymbol{q}' \in \mathcal{C}$ fitting E wrt $\mathcal{O}$ is equivalent to $\boldsymbol{q}$ wrt $\mathcal{O}$. (Equivalence of 2D-queries wrt an ontology is defined similarly to the 1D case.) If all $\boldsymbol{q} \in \mathcal{C}$ are uniquely characterised by some E wrt $\mathcal{O}$ within $\mathcal{C}' \supseteq \mathcal{C}$, we call $\mathcal{C}$ *uniquely characterisable wrt $\mathcal{O}$ within $\mathcal{C}'$*. Let $\mathcal{C}^n$ be the set of queries in $\mathcal{C}$ of temporal depth $\leq n$. We say that $\mathcal{C}$ is *polynomially characterisable wrt $\mathcal{O}$ for bounded temporal depth* if there is a polynomial f such that every $\boldsymbol{q} \in \mathcal{C}^n$ is characterised by some E of size $\leq f(n)$ within $\mathcal{C}^n$.

Note that $\Diamond \boldsymbol{r}$ is equivalent to $\bigcirc\Diamond_r \boldsymbol{r}$, so $\Diamond$ does not add any expressive power to $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ and $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q}) = \mathit{LTL}_p^{\bigcirc\Diamond_r}(\mathcal{Q})$; however, $\mathit{LTL}_p^{\bigcirc\Diamond}(\mathcal{Q}) \subsetneq \mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$

In the remainder of the paper, we investigate unique characterisability of 2D queries with respect to various query classes defined above.

4 $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$

The aim of this section is to give a criterion for (polynomial) unique characterisability of queries in the class $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ under certain conditions on the ontology and on $\mathcal{Q}$.

It will be convenient to represent queries $\boldsymbol{q}$ of the form (6) as a sequence

$$\boldsymbol{q} = \boldsymbol{r}_0(t_0), R_1(t_0, t_1), \dots, \boldsymbol{r}_{m-1}(t_{m-1}), R_m(t_{m-1}, t_m), \boldsymbol{r}_m(t_m), \tag{8}$$

where $R_i \in \{\mathit{suc}, <, \leq\}$, $\mathit{suc}(t, t')$ stands for $t' = t + 1$, and the t_i are variables over $(\mathbb{N}, <)$. We set $\mathit{var}(\boldsymbol{q}) = \{t_0, \dots, t_m\}$ (so we do not take into account the non-answer variables in the $\boldsymbol{r}_i$ here).

Example 4. Below are a 2D-query $\boldsymbol{q}$ and its representation of the form (8):

$$\exists P.B \land \bigcirc(\exists P.A \land \Diamond A) \;\rightsquigarrow\; \exists P.B(t_0), \mathit{suc}(t_0, t_1), \exists P.A(t_1), (t_1 < t_2), A(t_2)$$

with $\mathit{var}(\boldsymbol{q}) = \{t_0, t_1, t_2\}$.

We divide $\boldsymbol{q}$ of the form (8) into *blocks* $\boldsymbol{q}_i$ such that

$$\boldsymbol{q} = \boldsymbol{q}_0 \mathcal{R}_1 \boldsymbol{q}_1 \dots \mathcal{R}_n \boldsymbol{q}_n, \tag{9}$$

where $\mathcal{R}_i = R_1^i(t_0^i, t_1^i) \dots R_{n_i}^i(t_{n_i-1}^i, t_{n_i}^i)$ with $R_j^i \in \{<, \leq\}$ and

$$\boldsymbol{q}_i = \boldsymbol{r}_0^i(s_0^i)\mathit{suc}(s_0^i, s_1^i)\boldsymbol{r}_1^i(s_1^i) \dots \mathit{suc}(s_{k_i-1}^i, s_{k_i}^i)\boldsymbol{r}_{k_i}^i(s_{k_i}^i) \tag{10}$$

with $s_{k_i}^i = t_0^{i+1}$, $t_{n_i}^i = s_0^i$. If $k_i = 0$, the block $\boldsymbol{q}_i$ is called *primitive*.

Example 5. The query $\boldsymbol{q}$ from Example 4 has two blocks

$$\boldsymbol{q}_0 = \exists P.B(t_0), \mathit{suc}(t_0, t_1), \exists P.A(t_1) \quad \text{and} \quad \boldsymbol{q}_1 = A(t_2),$$

connected by $(t_1 < t_2)$. It contains one primitive block, $\boldsymbol{q}_1$.

Suppose we are given an ontology $\mathcal{O}$ and a class $\mathcal{Q}$ of domain queries. Then a primitive block $\boldsymbol{q}_i = \boldsymbol{r}_0^i(s_0^i)$ with $i > 0$ in $\boldsymbol{q}$ of the form (9) is called a *lone conjunct wrt $\mathcal{O}$ within $\mathcal{Q}$* if $\boldsymbol{r}_0^i$ is meet-reducible wrt $\mathcal{O}$ within $\mathcal{Q}$.

Example 6. Let $\mathcal{P}$ be the class of conjunctions of unary atoms; we call such queries *propositional*. The query $\Diamond A$, which is represented by the sequence $\top(t_0), (t_0 < t_1), A(t_1)$, does not have any lone conjuncts wrt the empty ontology within $\mathcal{Q}$, but A is a lone conjunct of $\Diamond A$ wrt $\mathcal{O} = \{A \equiv B \land C\}$ within $\mathcal{Q}$.

The query $\boldsymbol{q} = \Diamond A$ is uniquely characterised wrt the empty ontology within $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{P})$ by the example set $E = (E^+, E^-)$, where E^+ contains two temporal data instances $\emptyset, \{A\}$ and $\emptyset, \emptyset, \{A\}$ and E^- consists of one instance $\emptyset$. However, the same $\boldsymbol{q}$ cannot be uniquely characterised wrt $\mathcal{O} = \{A \equiv B \land C\}$ within $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{P})$ as shown in [22]. Observe also that A is a lone conjunct in $\boldsymbol{q}' = \Diamond(A \land \Diamond_r D)$ wrt $\mathcal{O}' = \mathcal{O} \cup \{D \sqsubseteq A\}$ but for the simplification $\boldsymbol{q}'' = \Diamond D$ of $\boldsymbol{q}'$ we have $\boldsymbol{q}'' \equiv_{\mathcal{O}'} \boldsymbol{q}'$ and $\boldsymbol{q}''$ does not have any lone conjuncts wrt $\mathcal{O}'$.

Example 6 shows that the notion of lone conjunct is dependent on the presentation of the query. To make lone conjuncts semantically meaningful, we introduce a normal form. Given an ontology $\mathcal{O}$ and a query $\boldsymbol{q}$ of the form (9), we say that $\boldsymbol{q}$ is in *normal form wrt $\mathcal{O}$* if the following conditions hold:

(n1) $\boldsymbol{r}_0^i \not\equiv_{\mathcal{O}} \top$ if $i > 0$, and $\boldsymbol{r}_{k_i}^i \not\equiv_{\mathcal{O}} \top$ if $i > 0$ or $k_i > 0$ (thus, of all the first/last $\boldsymbol{r}$ in a block only $\boldsymbol{r}_0^0$ can be trivial);

(n2) each $\mathcal{R}_i$ is either a single $t_0^i \leq t_1^i$ or a sequence of $<$;

(n3) $\boldsymbol{r}_{k_i}^i \not\models_{\mathcal{O}} \boldsymbol{r}_0^{i+1}$ if $\boldsymbol{q}_{i+1}$ is primitive and R_{i+1} is $\leq$;

(n4) $\boldsymbol{r}_0^{i+1} \not\models_{\mathcal{O}} \boldsymbol{r}_{k_i}^i$ if $i > 0$, $\boldsymbol{q}_i$ is primitive and R_{i+1} is $\leq$;

(n5) $\boldsymbol{r}_{k_i}^i(x) \land \boldsymbol{r}_0^{i+1}(x)$ is satisfiable wrt $\mathcal{O}$ whenever R_{i+1} is $\leq$.

Lemma 5. *Let $\mathcal{O}$ be an FO-ontology (possibly with $=$). Then every query $\boldsymbol{q} \in$ $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ is equivalent wrt $\mathcal{O}$ to a query in normal form of size at most $|\boldsymbol{q}|$ and of temporal depth at most the temporal depth of $\boldsymbol{q}$. This query can be computed in polynomial time if containment between queries in $\mathcal{Q}$ wrt $\mathcal{O}$ is decidable in polynomial time. For ELIQs, this is the case for DL-Lite$_{\mathcal{F}}$ but not for DL-Lite$_{\mathcal{H}}$ (unless* P = NP).

Proof. The transformation is straightforward: to ensure **(n1)**, drop any $\boldsymbol{r}_0^i$ and $\boldsymbol{r}_{k_i}^i$ for which **(n1)** fails and add one $<$ to the relevant $\mathcal{R}_i$. To ensure **(n2)**, replace any $\mathcal{R}_i$ containing at least one occurrence of $<$ with the sequence obtained from $\mathcal{R}_i$ by dropping all occurrences of $\leq$ and replace any $\mathcal{R}_i$ not containing any occurrence of $<$ by a single $\leq$. To ensure **(n3)**, drop any $\boldsymbol{r}_0^{i+1}$ with $\boldsymbol{r}_{k_i}^i \models_{\mathcal{O}} \boldsymbol{r}_0^{i+1}$ if $\boldsymbol{q}_{i+1}$ is primitive and $\mathcal{R}_{i+1}$ is $\leq$. To ensure **(n4)** drop any $\boldsymbol{r}_{k_i}^i$ with $\boldsymbol{r}_0^{i+1} \models_{\mathcal{O}} \boldsymbol{r}_{k_i}^i$ if $i > 0$, $\boldsymbol{q}_i$ is primitive and $\mathcal{R}_{i+1}$ is $\leq$. Finally, to ensure **(n5)**, replace $\leq$ by $<$ if $\boldsymbol{r}_{k_i}^i, \boldsymbol{r}_0^{i+1}$ are $\mathcal{O}$-compatible and $\mathcal{R}_{i+1}$ is $\leq$.

The second part follows from the fact that query containment in *DL-Lite*$_{\mathcal{F}}$ is in P and NP-hard in *DL-Lite*$_{\mathcal{H}}$ [32]. □

We call a query $\boldsymbol{q} \in$ $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ *safe wrt $\mathcal{O}$* if it is equivalent wrt $\mathcal{O}$ to an $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$-query in normal form that has no lone conjuncts.

It is to be noted that our temporal query languages do not admit containment reduction as, for example, there is no temporal data instance $\hat{\boldsymbol{q}}$ for $\boldsymbol{q} = \bigcirc(A \land \Diamond B)$ because it will have to fix the number of steps between 0 and the moment where B holds.

We are now in a position to formulate the main result of this section.

Theorem 15. *Suppose an ontology $\mathcal{O}$ admits singular$^+$characterisations within a class of domain queries $\mathcal{Q}$. Then the following hold*:

(i) *A query $\boldsymbol{q} \in$ $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ is uniquely characterisable within $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ wrt $\mathcal{O}$ iff $\boldsymbol{q}$ is safe wrt $\mathcal{O}$.*

(ii) *If $\mathcal{O}$ admits polynomial-size singular$^+$ characterisations within $\mathcal{Q}$, then those queries that are uniquely characterisable within $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ are actually polynomially characterisable within $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$.*

(iii) *The class $LTL_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ is polynomially characterisable for bounded temporal depth if $\mathcal{O}$ admits polynomial-size singular$^+$characterisations within $\mathcal{Q}$.*

(iv) *The class $LTL_p^{\bigcirc\Diamond}(\mathcal{Q})$ is uniquely characterisable. It is polynomially characterisable if $\mathcal{O}$ admits polynomial-size singular$^+$characterisations within $\mathcal{Q}$.*

To prove Theorem 15 we first provide some notation for talking about the entailment relation $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}$. Let $\mathcal{D} = \mathcal{A}_0, \dots, \mathcal{A}_n$, $a \in ind(\mathcal{D})$, and let $\boldsymbol{q}$ take the form (8). A map $h\colon var(\boldsymbol{q}) \to [0, \max(\mathcal{D})]$ is called a *root $\mathcal{O}$-homomorphism* from $\boldsymbol{q}$ to $(\mathcal{D}, a)$ if $h(t_0) = 0$, $\mathcal{O}, \mathcal{A}_{h(t)} \models \boldsymbol{r}(a)$ if $\boldsymbol{r}(t) \in \boldsymbol{q}$, $h(t') = h(t) + 1$ if $suc(t, t') \in \boldsymbol{q}$, and $h(t)\, R\, h(t')$ if $R(t, t') \in \boldsymbol{q}$ for $R \in \{<, \leq\}$. It is readily seen that $\mathcal{O}, \mathcal{D}, a, 0 \models \boldsymbol{q}$ iff there exists a root $\mathcal{O}$-homomorphism from $\boldsymbol{q}$ to $(\mathcal{D}, a)$.

Let $b \geq 1$. The instance $\mathcal{D}$ is said to be *b-normal wrt $\mathcal{O}$* if it takes the form

$$\mathcal{D} = \mathcal{D}_0 \emptyset^b \mathcal{D}_1 \dots \emptyset^b \mathcal{D}_n, \text{ where } \mathcal{D}_i = \mathcal{A}_0^i \dots \mathcal{A}_{k_i}^i, \tag{11}$$

with $b > k_i \geq 0$ and $\mathcal{A}_0^i \not\equiv_{\mathcal{O}} \emptyset$ if $i > 0$, and $\mathcal{A}_{k_i}^i \not\equiv_{\mathcal{O}} \emptyset$ if $i > 0$ or $k_i > 0$ (thus, of all the first/last $\mathcal{A}$ in a $\mathcal{D}_i$ only $\mathcal{A}_0^0$ can be trivial). Following the terminology for queries, we call each $\mathcal{D}_i$ *a block of* $\mathcal{D}$. For a block $\mathcal{D}_i$ in $\mathcal{D}$, we denote by $I(\mathcal{D}_i)$ the subset of $[0, \max(\mathcal{D})]$ occupied by $\mathcal{D}_i$. Then we call a root $\mathcal{O}$-homomorphism $h\colon \boldsymbol{q} \to \mathcal{D}$ *block surjective* if every $j \in I(\mathcal{D}_i)$ with a block $\mathcal{D}_i$ is in the range $ran(h)$ of h. We next aim to state that after 'weakening' the non-temporal data instances in blocks, no root $\mathcal{O}$-homomorphism from $\boldsymbol{q}$ to $\mathcal{D}$ exists. This is only required if the non-temporal data instances are obtained from queries in $\mathcal{Q}$, and so we express 'weakening' using singular$^+$characterisations of queries in $\mathcal{Q}$ wrt $\mathcal{O}$.

In detail, suppose $\mathcal{A}_j^i = \hat{\boldsymbol{s}}_j^i$, for queries $\boldsymbol{s}_j^i \in \mathcal{Q}$. Given $\boldsymbol{s} \in \mathcal{Q}$, take the singular$^+$characterisation $(\{\hat{\boldsymbol{s}}\}, \mathcal{N}_{\boldsymbol{s}})$ of $\boldsymbol{s}$ wrt $\mathcal{O}$ within $\mathcal{Q}$, where $\{\hat{\boldsymbol{s}}\}$ is the only positive data instance, and $\mathcal{N}_{\boldsymbol{s}}$ is the set of negative data instances. To make our notation more uniform, we think of the pointed data instances in $\mathcal{N}_{\boldsymbol{s}}$ as having the form $\hat{\boldsymbol{s}}'$, for a suitable CQ $\boldsymbol{s}'$ (which is not necessarily in $\mathcal{Q}$).

For $\ell \in \mathit{ran}(h)$, let

$$\boldsymbol{r}_\ell = \bigwedge_{\boldsymbol{r}(t) \in \boldsymbol{q}, h(t) = \ell} \boldsymbol{r}.$$

Then h is called *data surjective* if $\mathcal{O}, \hat{\boldsymbol{s}} \not\models \boldsymbol{r}_\ell(a)$, for any $\boldsymbol{s} \in \mathcal{N}_{\boldsymbol{s}^i_j}$ and any $\ell \in \mathit{ran}(h)$ such that $\boldsymbol{s}^i_j \not\equiv \top$, where $\hat{\boldsymbol{s}}^i_j$ is the data instance placed at ℓ in $\mathcal{D}$.

We call the root $\mathcal{O}$-homomorphism $h : \boldsymbol{q} \to \mathcal{D}$ a *root $\mathcal{O}$-isomorphism* if it is data surjective and, for the blocks $\boldsymbol{q}_0, \dots, \boldsymbol{q}_m$ of $\boldsymbol{q}$, we have $n = m$ and h restricted to $\mathit{var}(\boldsymbol{q}_i)$ is a bijection onto $I(\mathcal{D}_i)$ for all $i \leq n$ (in particular, h is block surjective). Intuitively, if we have a root $\mathcal{O}$-isomorphism $h \colon \boldsymbol{q} \to \mathcal{D}$, then $\boldsymbol{q}$ is almost the same as $\mathcal{D}$ except for differences between the sequences $\mathcal{R}_i$ in $\boldsymbol{q}$ and the gaps between blocks in $\mathcal{D}$.

Let $\mathcal{Q}$, $\mathcal{O}$, $\boldsymbol{q}$, and $\mathcal{D}$ be as before with $\mathcal{D}$ of the form (11), where $\mathcal{A}^i_j = \hat{\boldsymbol{s}}^i_j$, for $\boldsymbol{s}^i_j \in \mathcal{Q}$. The following rules will be used to define the negative examples in the unique characterisation of $\boldsymbol{q}$ and as steps in the learning algorithm. They are applied to $\mathcal{D}$:

(a) replace some $\hat{\boldsymbol{s}}^i_j$ with $\boldsymbol{s}^i_j \not\equiv_{\mathcal{O}} \top$ by an $\hat{\boldsymbol{s}} \in \mathcal{N}_{\boldsymbol{s}^i_j}$, for i, j such that $(i, j) \neq (0, 0)$—that is, the rule is not applied to $\boldsymbol{s}^0_0$;

(b) replace some pair $\hat{\boldsymbol{s}}^i_j \hat{\boldsymbol{s}}^i_{j+1}$ within block i by $\hat{\boldsymbol{s}}^i_j \emptyset^b \hat{\boldsymbol{s}}^i_{j+1}$;

(c) replace some $\hat{\boldsymbol{s}}^i_j$ with $\boldsymbol{s}^i_j \not\equiv_{\mathcal{O}} \top$ by $\hat{\boldsymbol{s}}^i_j \emptyset^b \hat{\boldsymbol{s}}^i_j$, where $k_i > j > 0$—that is, the rule is not applied to $\boldsymbol{s}^i_j$ if it is on the border of its block;

(d) replace $\hat{\boldsymbol{s}}^i_{k_i}$ ($k_i > 0$) by $\hat{\boldsymbol{s}} \emptyset^b \hat{\boldsymbol{s}}^i_{k_i}$, for some $\hat{\boldsymbol{s}} \in \mathcal{N}_{\boldsymbol{s}^i_{k_i}}$, or replace $\hat{\boldsymbol{s}}^i_0$ ($k_i > 0$) by $\hat{\boldsymbol{s}}^i_0 \emptyset^b \hat{\boldsymbol{s}}$, for some $\hat{\boldsymbol{s}} \in \mathcal{N}_{\boldsymbol{s}^i_0}$;

(e) replace $\hat{\boldsymbol{s}}^0_0$ with $\boldsymbol{s}^0_0 \not\equiv_{\mathcal{O}} \top$ by $\hat{\boldsymbol{s}} \emptyset^b \hat{\boldsymbol{s}}^0_0$, for $\hat{\boldsymbol{s}} \in \mathcal{N}_{\boldsymbol{s}^0_0}$, if $k_0 = 0$, and by $\hat{\boldsymbol{s}}^0_0 \emptyset^b \hat{\boldsymbol{s}}^0_0$ if $k_0 > 0$.

If $k_i = 0$, $i > 0$, and $\boldsymbol{s}^i_0$ is meet-reducible wrt $\mathcal{O}$ within $\mathcal{Q}$, then we say that $\boldsymbol{s}^i_0$ is a *lone conjunct wrt $\mathcal{O}$ within $\mathcal{Q}$* in $\mathcal{D}$.

Lemma 6. *Assume $\mathcal{Q}, \mathcal{O}$, and $\boldsymbol{q}$ are as above. Let b exceed the number of $\Diamond$ and $\bigcirc$ in $\boldsymbol{q}$, let $\boldsymbol{q}$ be in normal form, and let $\mathcal{D}$ be b-normal without lone conjuncts wrt $\mathcal{O}$ within $\mathcal{Q}$. If $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}$ but $\mathcal{O}, \mathcal{D}' \not\models \boldsymbol{q}$, for any $\mathcal{D}'$ obtained from $\mathcal{D}$ by applying any of the rules (a)–(e), then any root $\mathcal{O}$-homomorphim $h \colon \boldsymbol{q} \to \mathcal{D}$ is a root $\mathcal{O}$-isomorphism.*

Proof. We assume that $\boldsymbol{q}$ is constructed using $\boldsymbol{r}^i_j \in \mathcal{Q}$ and $\mathcal{D}$ is constructed using $\boldsymbol{s}^i_j \in \mathcal{Q}$. Let $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}$. Take a root $\mathcal{O}$-homomorphism $h \colon \boldsymbol{q} \to \mathcal{D}$.

Suppose first that h is not block surjective. Since $h(0) = 0$, we find i, j with $(i, j) \neq (0, 0)$ such that the time point of $\hat{\boldsymbol{s}}^i_j$ is not in the range of h. If $\hat{\boldsymbol{s}}^i_j$ is not on the border of its block, that is $0 < j < k_i$, then we obtain from h a root $\mathcal{O}$-homomorphism into the data instance $\mathcal{D}'$ obtained from $\mathcal{D}$ by rule (b) and have derived a contradiction. If $0 = j$ or $k_i = j$, then we obtain from h a root $\mathcal{O}$-homomorphism into the data instance $\mathcal{D}'$ obtained from $\mathcal{D}$ by rule (a) applied to $\boldsymbol{s}^i_j$ and have derived a contradiction.

Assume next that h is block surjective but not data surjective. Then we find $\ell \in \mathit{ran}(h)$ with $\hat{\boldsymbol{s}}^i_j$ such that $\boldsymbol{s}^i_j \not\equiv_{\mathcal{O}} \top$ placed at ℓ such that $\mathcal{O}, \hat{\boldsymbol{s}} \models \boldsymbol{r}_\ell(a)$ for some $\boldsymbol{s} \in \mathcal{N}_{\boldsymbol{s}^i_j}$. But then h is a root $\mathcal{O}$-homomorphism into the data instance $\mathcal{D}'$ obtained from $\mathcal{D}$ by applying rule (a) to $\boldsymbol{s}^i_j$, that is, replacing $\hat{\boldsymbol{s}}^i_j$ by $\hat{\boldsymbol{s}}$ with $\boldsymbol{s} \in \mathcal{N}_{\boldsymbol{s}^i_j}$.

Suppose now that $h \colon \boldsymbol{q} \to \mathcal{D}$ is a block and data surjective, $(t \leq t') \in \boldsymbol{q}$ and $h(t) = h(t') = \ell$ lies in the ith block of $\mathcal{D}$. Then $h^{-1}(\ell) = \{t_1, \dots, t_k\}$ with $k \geq 2$ and $(t_j \leq t_{j+1}) \in \boldsymbol{q}$, $1 \leq j < k$. Let $\boldsymbol{r}_1, \dots, \boldsymbol{r}_k$ be the queries with $\boldsymbol{r}_j(t_j)$ in $\boldsymbol{q}$. As $\boldsymbol{q}$ satisfies **(n1)** and **(n2)**, there is j with $\boldsymbol{r}_j \not\equiv_{\mathcal{O}} \top$. Hence, $\boldsymbol{s}^i_{j_0} \not\equiv_{\mathcal{O}} \top$ for the query $\boldsymbol{s}^i_{j_0}$ with $\hat{\boldsymbol{s}}^i_{j_0}$ at ℓ in $\mathcal{D}$. Moreover, by data surjectivity, $\boldsymbol{r}_1 \wedge_{\mathcal{O}} \dots \wedge_{\mathcal{O}} \boldsymbol{r}_k \equiv_{\mathcal{O}} \boldsymbol{s}^i_{j_0}$. Consider possible locations of j_0 in its block.

Case 1: j_0 has both a left and a right neighbour in its block. Then there is $\mathcal{D}'$ obtained by (c)—i.e., by replacing $\hat{\boldsymbol{s}}^i_{j_0}$ with $\hat{\boldsymbol{s}}^i_{j_0} \emptyset^b \hat{\boldsymbol{s}}^i_{j_0}$—and a root $\mathcal{O}$-homomorphism $h' \colon \boldsymbol{q} \to \mathcal{D}'$, which 'coincides' with h except that $h'(t_1)$ is the point with the first $\hat{\boldsymbol{s}}^i_{j_0}$ and $h'(t_j)$, for $j = 2, \dots, k$, is the point with the second $\hat{\boldsymbol{s}}^i_{j_0}$.

Case 2: j_0 has no neighbours in its block and $i \neq 0$, so this block is primitive and $\boldsymbol{s}^i_{j_0}$ is not equivalent to a conjunction of queries as $\mathcal{D}$ has no lone conjuncts by our assumption. Observe that the blocks of $t_1, \dots, t_k$ are all different and primitive. As $\boldsymbol{s}^i_{j_0}$ is not equivalent to a conjunction of queries, we have

$$\boldsymbol{r}_1 \equiv_{\mathcal{O}} \dots \equiv_{\mathcal{O}} \boldsymbol{r}_k \equiv_{\mathcal{O}} \boldsymbol{s}^i_{j_0}.$$

However, $\boldsymbol{r}_j \not\equiv_{\mathcal{O}} \boldsymbol{r}_{j+1}$ by **(n3)** and **(n4)**. Thus, Case 2 cannot happen.

Case 3: j_0 has a left neighbour in its block but no right neighbour. Then $\boldsymbol{r}_1 \not\models_{\mathcal{O}} \boldsymbol{r}_2$ in view of **(n3)**, and so $\boldsymbol{r}_1 \not\models_{\mathcal{O}} \boldsymbol{s}^i_{j_0}$. As $\boldsymbol{s}^i_{j_0} \models_{\mathcal{O}} \boldsymbol{r}_1$, there is $\hat{\boldsymbol{s}} \in \mathcal{N}_{\boldsymbol{s}^i_{j_0}}$ with $\mathcal{O}, \hat{\boldsymbol{s}} \not\models_{\mathcal{O}} \boldsymbol{r}_1$. Let $\mathcal{D}'$ be obtained by the first part of (d) by replacing $\hat{\boldsymbol{s}}^i_{j_0}$ with $\hat{\boldsymbol{s}}\emptyset^b\hat{\boldsymbol{s}}^i_{j_0}$. Then there is a root $\mathcal{O}$-homomorphism $h' \colon \boldsymbol{q} \to \mathcal{D}'$ that sends t_1 to the point of $\boldsymbol{s}$ and the remaining t_j to the point of $\boldsymbol{s}^i_{j_0}$.

Case 4: j_0 has a right neighbour in its block, $i \neq 0$, and it has no left neighbour. This case is dual to Case 3 and we use the second part of (d).

Case 5: $i = 0$ and $j_0 = 0$. If block 0 is primitive, then all of the $\boldsymbol{r}_i(t_i)$ are primitive blocks in $\boldsymbol{q}$. By **(n3)**, $\boldsymbol{r}_1 \not\models_{\mathcal{O}} \boldsymbol{r}_2$, and so $\boldsymbol{r}_1 \not\models_{\mathcal{O}} \boldsymbol{s}^i_{j_0}$. Take $\hat{\boldsymbol{s}} \in \mathcal{N}_{\boldsymbol{s}^i_{j_0}}$. By the first part of (e), we have $\mathcal{D}'$ obtained by replacing $\hat{\boldsymbol{s}}^0_0$ with $\hat{\boldsymbol{s}}\emptyset^b\hat{\boldsymbol{s}}^0_0$. Then there is a root $\mathcal{O}$-homomorphism $h' \colon \boldsymbol{q} \to \mathcal{D}'$ that sends t_1 to the point of $\boldsymbol{s}$ and the remaining t_j to the point of $\boldsymbol{s}^0_0$.

Finally, if block 0 is not primitive, the second part of (e) gives $\mathcal{D}'$ by replacing $\hat{\boldsymbol{s}}^0_0$ in $\mathcal{D}$ with $\hat{\boldsymbol{s}}^0_0\emptyset^b\hat{\boldsymbol{s}}^0_0$. We obtain a root $\mathcal{O}$-homomorphism from $\boldsymbol{q}$ to $\mathcal{D}'$ by sending t_1 to the first $\hat{\boldsymbol{s}}^0_0$ and the remaining t_j to the second $\hat{\boldsymbol{s}}^0_0$. ☐

We can now prove Theorem 15. Suppose an ontology $\mathcal{O}$ admits singular$^+$ characterisations within a class of domain queries $\mathcal{Q}$. Let $\boldsymbol{q} \in \mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ in normal form wrt $\mathcal{O}$ take the form (9) with $\boldsymbol{q}_i$ of the form (10). We define an example set $E = (E^+, E^-)$ characterising $\boldsymbol{q}$ under the assumption that $\boldsymbol{q}$ has no lone conjuncts wrt $\mathcal{O}$. Let b be the number of $\bigcirc$ and $\Diamond$ in $\boldsymbol{q}$ plus 1. For every block $\boldsymbol{q}_i$ of the form (10), let $\hat{\boldsymbol{q}}_i$ be the temporal data instance

$$\hat{\boldsymbol{q}}_i = \hat{\boldsymbol{r}}^i_0\hat{\boldsymbol{r}}^i_1 \dots \hat{\boldsymbol{r}}^i_{k_i}.$$

For any two blocks $\boldsymbol{q}_i, \boldsymbol{q}_{i+1}$ such that $\boldsymbol{r}^i_{k_i}$ and $\boldsymbol{r}^{i+1}_0$ are $\mathcal{O}$-compatible, we take the temporal data instance

$$\hat{\boldsymbol{q}}_i \bowtie \hat{\boldsymbol{q}}_{i+1} = \hat{\boldsymbol{r}}^i_0 \dots \hat{\boldsymbol{r}}^i_{k_i-1}\widehat{\boldsymbol{r}^i_{k_i} \wedge \boldsymbol{r}^{i+1}_0}\hat{\boldsymbol{r}}^{i+1}_1 \dots \hat{\boldsymbol{r}}^{i+1}_{k_{i+1}}.$$

Now, the set E^+ contains the data instances given by

- $\mathcal{D}_b = \bar{\boldsymbol{q}}_0\emptyset^b \dots \bar{\boldsymbol{q}}_i\emptyset^b\bar{\boldsymbol{q}}_{i+1} \dots \emptyset^b\bar{\boldsymbol{q}}_n$,
- $\mathcal{D}_i = \bar{\boldsymbol{q}}_0\emptyset^b \dots (\bar{\boldsymbol{q}}_i \bowtie \bar{\boldsymbol{q}}_{i+1}) \dots \emptyset^b\bar{\boldsymbol{q}}_n$ if $\mathcal{R}_{i+1}$ is $\leq$,
- $\mathcal{D}_i = \bar{\boldsymbol{q}}_0\emptyset^b \dots \bar{\boldsymbol{q}}_i\emptyset^{n_{i+1}}\bar{\boldsymbol{q}}_{i+1} \dots \emptyset^b\bar{\boldsymbol{q}}_n$ otherwise.

Here, $\emptyset^b$ is a sequence of b-many $\emptyset$ and similarly for $\emptyset^{n_{i+1}}$. The set E^- contains all data instances of the form

- $\mathcal{D}^-_i = \bar{\boldsymbol{q}}_0\emptyset^b \dots \bar{\boldsymbol{q}}_i\emptyset^{n_{i+1}-1}\bar{\boldsymbol{q}}_{i+1} \dots \emptyset^b\bar{\boldsymbol{q}}_n$ if $n_{i+1} > 1$,
- $\mathcal{D}^-_i = \bar{\boldsymbol{q}}_0\emptyset^b \dots \bar{\boldsymbol{q}}_i \bowtie \bar{\boldsymbol{q}}_{i+1} \dots \emptyset^b\bar{\boldsymbol{q}}_n$ if $\mathcal{R}_{i+1}$ is a single $<$ and $\boldsymbol{r}^i_{k_i}$ and $\boldsymbol{r}^{i+1}_0$ are $\mathcal{O}$-compatible,
- the data instances obtained from $\mathcal{D}_b$ by applying to it each of the rules (a)–(e) in all possible ways exactly once.

We show that E characterises $\boldsymbol{q}$. Clearly, $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}$ for all $\mathcal{D} \in E^+$. To establish $\mathcal{O}, \mathcal{D} \not\models \boldsymbol{q}$ for $\mathcal{D} \in E^-$, we need the following:

Claim 1. (i) *There is only one root $\mathcal{O}$-homomorphism $h \colon \boldsymbol{q} \to \mathcal{D}_b$, and it maps isomorphically each var($\boldsymbol{q}_i$) onto $I(\bar{\boldsymbol{q}}_i)$.*

(ii) *$\mathcal{O}, \mathcal{D}^-_i \not\models \boldsymbol{q}$, for any $\mathcal{R}_i$ different from $\leq$.*

(iii) *If $\mathcal{D}'_b$ is obtained from $\mathcal{D}_b$ by replacing some $\bar{\boldsymbol{q}}_i$ with $\bar{\boldsymbol{q}}'_i$ such that $\mathcal{O}, \bar{\boldsymbol{q}}'_i, \ell \not\models \boldsymbol{q}_i$ for any $\ell \leq \max(\bar{\boldsymbol{q}}'_i)$, then $\mathcal{O}, \mathcal{D}'_b \not\models \boldsymbol{q}$. In particular, $\mathcal{O}, \mathcal{D} \not\models \boldsymbol{q}$, for all $\mathcal{D} \in E^-$.*

Proof of claim. (i) Let h be a root $\mathcal{O}$-homomorphism. As $\boldsymbol{q}$ is in normal form and the gaps between $\bar{\boldsymbol{q}}_i$ and $\bar{\boldsymbol{q}}_{i+1}$ are not shorter than any block in $\boldsymbol{q}$, every *var*($\boldsymbol{q}_i$), where $\boldsymbol{q}_i$ is a block in $\boldsymbol{q}$, is mapped by h to a single $I(\bar{\boldsymbol{q}}_j)$, where $\bar{\boldsymbol{q}}_j$ is a block of D_b. Hence we can define a function $f \colon [0, n] \to [0, n]$ by setting $f(i) = j$ if $f(\mathit{var}(\boldsymbol{q}_i)) \subseteq I(\bar{\boldsymbol{q}}_j)$. Observe that $f(0) = 0$ and $i < j$ implies $f(i) \leq f(j)$. It also follows from the definition of the normal form that if $f(i) = i$, then h isomorphically maps *var*($\boldsymbol{q}_i$) onto $I(\bar{\boldsymbol{q}}_i)$ and $f(i-1) < i$ and $f(i+1) > i$ (observe that here $(\mathbf{n3})$ and $(\mathbf{n4})$ are required as they prohibit that *var*($\boldsymbol{q}_i$) and *var*($\boldsymbol{q}_{i+1}$) are merged if $R_{i+1} = \leq$ and *var*($\boldsymbol{q}_i$) or *var*($\boldsymbol{q}_{i+1}$) are a singleton). It remains to show that $f(i) = i$ for all i.

We first observe that $f(1) \geq 1$ and $f(j) = j$, for $j = \max\{i \mid f(i) \geq i\}$, from which again $f(j-1) < j$ and $f(j+1) > j$. Then we can proceed in the same way inductively by considering h and f restricted to the smaller intervals $[j, n]$ and $[0, j]$.

(ii) Suppose $\mathcal{R}_i$ is not $\leq$ but there is a root $\mathcal{O}$-homomorphism $h\colon \boldsymbol{q} \to \mathcal{D}_i^-$. Consider the location of $h(s_0^i) = \ell$. One can show similarly to (i) that $\ell \in I(\bar{\boldsymbol{q}}_j)$ for some $j \geq i$. Since $\boldsymbol{r}_{k_{i+1}}^{i+1} \not\equiv_{\mathcal{O}} \top$ and by the construction of $\mathcal{D}_i^-$, $h(s_0^{i+1})$ lies in some $I(\bar{\boldsymbol{q}}_j)$ with $j > i+1$. But then there is a root $\mathcal{O}$-homomorphism $h'\colon \boldsymbol{q} \to \mathcal{D}_b$ different from the one in (i), which is impossible.

(iii) is proved analogously. This completes the proof of the claim.

Now assume that $\boldsymbol{q}' \in \mathcal{Q}$ in normal form is given and $\boldsymbol{q}' \not\equiv_{\mathcal{O}} \boldsymbol{q}$. We have to show that $\boldsymbol{q}'$ does not fit E. If $\mathcal{O}, \mathcal{D}_b \not\models \boldsymbol{q}'$, we are done as $\mathcal{D}_b \in E^+$. Otherwise, let h be a root $\mathcal{O}$-homomorphism witnessing $\mathcal{O}, \mathcal{D}_b \models \boldsymbol{q}'$. If h is not a root $\mathcal{O}$-isomorphism, then by Lemma 6, there exists a data instance $\mathcal{D}$ obtained from $\mathcal{D}_b$ by applying one of the rules (a)–(e) such that $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}'$. As $\mathcal{D} \in E^-$, we are done.

So suppose $h\colon \boldsymbol{q}' \to \mathcal{D}_b$ is a root $\mathcal{O}$-isomorphism. Then the difference between $\boldsymbol{q}'$ and $\boldsymbol{q}$ can only be in the sequences of $\Diamond$ and $\Diamond_r$ between blocks. To be more precise, $\boldsymbol{q}$ is of the form (9),

$$\boldsymbol{q}' = \boldsymbol{q}_0 \mathcal{R}_1' \boldsymbol{q}_1 \dots \mathcal{R}_n' \boldsymbol{q}_n \tag{12}$$

and $\mathcal{R}_i \neq \mathcal{R}_i'$ for some i. Four cases are possible:

- $\mathcal{R}_i = (r_0 \leq r_1)$ and $\mathcal{R}_i' = (s_0 < s_1) \dots (s_{l-1} < s_l)$, for $l \geq 1$. In this case, $\mathcal{O}, \mathcal{D}_i \not\models \boldsymbol{q}'$, for $\mathcal{D}_i \in E^+$.
- $\mathcal{R}_i = (r_0 < r_1) \dots (r_{k-1} < r_k)$, $\mathcal{R}_i' = (s_0 < s_1) \dots\ (s_{l-1} < s_l)$, for $l > k$. Then again $\mathcal{O}, \mathcal{D}_i \not\models \boldsymbol{q}'$.
- $\mathcal{R}_i = (r_0 < r_1) \dots (r_{k-1} < r_k)$, $\mathcal{R}_i' = (s_0 \leq s_1)$, for $k \geq 1$. In this case $\mathcal{O}, \mathcal{D}_i^- \models \boldsymbol{q}'$, for $\mathcal{D}_i^- \in E^-$. (Note that the compatibility condition is satisfied as $\boldsymbol{q}'$ is in normal form.)
- $\mathcal{R}_i = (r_0 < r_1) \dots (r_{k-1} < r_k)$ and $\mathcal{R}_i' = (s_0 < s_1) \dots\ (s_{l-1} < s_l)$, for $l < k$. Then again $\mathcal{O}, \mathcal{D}_i^- \models \boldsymbol{q}'$.

We now show the converse direction in Theorem 15 (i). Suppose $\boldsymbol{q}$ in normal form (9) does contain a lone conjunct $\boldsymbol{q}_i = \boldsymbol{r}$ wrt $\mathcal{O}$ within $\mathcal{Q}$. Let $\boldsymbol{r}^-$ be the last query of the block $\boldsymbol{q}_{i-1}$ and let $\boldsymbol{r}^+$ be the first query of the block $\boldsymbol{q}_{i+1}$.

Now let $\boldsymbol{r} \equiv_{\mathcal{O}} \boldsymbol{r}_1 \wedge \boldsymbol{r}_2$ and $\boldsymbol{r}_i \not\models_{\mathcal{O}} \boldsymbol{r}$, $i = 1, 2$. Observe that, for $\boldsymbol{s} \in \{\boldsymbol{r}^-, \boldsymbol{r}^+\}$,

- $\boldsymbol{s}$ and $\boldsymbol{r}_i$ are compatible w.r.t. $\mathcal{O}$ if $\boldsymbol{r}^-$ and $\boldsymbol{r}$ are compatible w.r.t $\mathcal{O}$;
- if $\boldsymbol{s} \not\models_{\mathcal{O}} \boldsymbol{r}$, then $\boldsymbol{s} \not\models_{\mathcal{O}} \boldsymbol{r}_1$ or $\boldsymbol{s} \not\models_{\mathcal{O}} \boldsymbol{r}_2$.

Hence one of the queries $\boldsymbol{s}_1'$ or $\boldsymbol{s}_1''$ below is in normal form:

$$\boldsymbol{s}_1' = \boldsymbol{q}_0 \mathcal{R}_1 \dots \mathcal{R}_i \boldsymbol{s}_1 (\leq) \boldsymbol{s}_2 \mathcal{R}_{i+1} \dots \mathcal{R}_n \boldsymbol{q}_n,$$
$$\boldsymbol{s}_2' = \boldsymbol{q}_0 \mathcal{R}_1 \dots \mathcal{R}_i \boldsymbol{s}_1 (\leq) \boldsymbol{s}_2 (\leq) \boldsymbol{s}_1 \mathcal{R}_{i+1} \dots \mathcal{R}_n \boldsymbol{q}_n,$$

where $\{\boldsymbol{s}_1, \boldsymbol{s}_2\} = \{r_1, \boldsymbol{r}_2\}$. Pick one of $\boldsymbol{s}_1'$ and $\boldsymbol{s}_2'$, which is in normal form, and denote it by $\boldsymbol{s}_1'$. For $n \geq 2$, let $\boldsymbol{s}_n'$ be the query obtained from $\boldsymbol{s}_1'$ by duplicating n times the part $\boldsymbol{s}_1(\leq)\boldsymbol{s}_2$ in $\boldsymbol{s}_1'$ and inserting $\leq$ between the copies. It is readily seen that $\boldsymbol{s}_n'$ is in normal form. Clearly, $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{s}_n'$ and, similarly to the proof of Claim 1, one can show that $\boldsymbol{s}_n' \not\models_{\mathcal{O}} \boldsymbol{q}$, for any $n \geq 1$.

Suppose $E = (E^+, E^-)$ characterises $\boldsymbol{q}$ and $n = \max\{\max(\mathcal{D}) \mid \mathcal{D} \in E^-\} + 1$. Then there exists $\mathcal{D} \in E^-$ with $\mathcal{O}, \mathcal{D} \models \boldsymbol{s}_n'$, so we have a root $\mathcal{O}$-homomorphism $h\colon \boldsymbol{s}_n' \to \mathcal{D}$. By the pigeonhole principle, h maps some variables of the queries $\boldsymbol{s}_1, \boldsymbol{s}_2$ in $\boldsymbol{s}_n'$ to the same point in $\mathcal{D}$. But then h can be readily modified to obtain a root $\mathcal{O}$-homomorphism $h'\colon \boldsymbol{q} \to \mathcal{D}$, which is a contradiction. This finishes the proof of (i).

(ii) follows from the proof of (i) as (E^+, E^-) is of polynomial size if the singular$^+$characterisations are of polynomial size.

(iii) We aim to characterise $\boldsymbol{q}$ in normal form (9), which may contain lone conjuncts wrt $\mathcal{O}$ within $\mathcal{Q}$ in the class of queries from $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{Q})$ of temporal depth at most $n = \mathit{tdp}(\boldsymbol{q})$. We first observe a variation of Lemma 6. Extend the rules (a)–(e) by the following rule: if $\hat{\boldsymbol{s}}$ is a block in $\mathcal{D}$ with $\boldsymbol{s}$ a lone conjunct in $\mathcal{D}$, then let $\mathcal{N}_{\boldsymbol{q}} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_k\}$ with $\boldsymbol{s}_i \not\equiv_{\mathcal{O}} \boldsymbol{s}_j$, for $i \neq j$, and

(f$_n$) replace $\boldsymbol{s}$ with $(\boldsymbol{s}_1 \emptyset^b \cdots \emptyset^b \boldsymbol{s}_k)^n$.

By Lemma 3 (ii), $|\mathcal{N}_{\boldsymbol{q}}| \geq 2$. Now Lemma 6 still holds if we admit lone conjuncts in $\mathcal{D}$ but only consider $\boldsymbol{q}$ with at most n blocks and add rule (f$_n$) to (a)–(e). To see this, one only has to modify the argument for Case 2 in a straightforward way. With the above modification of Lemma 6, we continue as follows. The set E^+ of positive examples is defined as before. The set E^- of negative examples is defined by adding to the set E^- defined under (i) the results of applying (f_n) to $\mathcal{D}_b$ in all possible ways exactly once.

For the proof that (E^+, E^-) characterises $\boldsymbol{q}$ within the class of queries of temporal depth at most n, observe that $\mathcal{O}, \mathcal{D}' \not\models \boldsymbol{q}$ for the data instance $\mathcal{D}'$ obtained from $\mathcal{D}_b$ by applying (f_n).

(iv) Assume $\boldsymbol{q} \in \mathit{LTL}_p^{\bigcirc\Diamond}(\mathcal{Q})$ is given. The proof of (i) shows that (E^+, E^-), defined in the same way as in (i), characterises $\boldsymbol{q}$ wrt $\mathcal{O}$ within $\mathit{LTL}_p^{\bigcirc\Diamond}(\mathcal{Q})$ even if $\boldsymbol{q}$ contains lone conjuncts: the proof of Lemma 6 becomes much simpler as any block and type surjective root $\mathcal{O}$-homomorphism h is now a root $\mathcal{O}$-isomorphism. Note that therefore rules (c), (d), and (e) are not needed. This completes the proof of Theorem 15.

As an immediate consequence of Lemma 1 and Theorems 9, 12 and 15 we obtain:

Theorem 16. (i) *For any DL-Lite$_\mathcal{H}$ or DL-Lite$_\mathcal{F}^-$ ontology $\mathcal{O}$,*

- *a 2D-query $\boldsymbol{q} \in \mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{ELI})$ is uniquely characterisable—in fact, polynomially characterisable—wrt $\mathcal{O}$ within $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{ELI})$ iff $\boldsymbol{q}$ is safe wrt $\mathcal{O}$*;
- *$\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}(\mathcal{ELI})$ is polynomially characterisable wrt $\mathcal{O}$ for bounded temporal depth*;
- *$\mathit{LTL}_p^{\bigcirc\Diamond}(\mathcal{Q})$ is polynomially characterisable wrt $\mathcal{O}$.*

(ii) *Let σ be a signature. Every σ-ELIQ $\boldsymbol{q}$ is uniquely characterisable wrt any $\mathcal{ALCHI}$-ontology $\mathcal{O}$ in σ within the class of σ-ELIQs.*

5 $\mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{Q}^\sigma)$

Let $\mathcal{Q}$ be a domain query language and σ a finite signature of predicate symbols. Then $\mathcal{Q}^\sigma$ denotes the set of queries in $\mathcal{Q}$ that only use symbols in σ. We remind the reader that $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{Q}^\sigma)$ is the class of temporal path queries of the form

$$\boldsymbol{q} = \boldsymbol{r}_0 \wedge (\boldsymbol{l}_1 \mathbin{\mathsf{U}} (\boldsymbol{r}_1 \wedge (\boldsymbol{l}_2 \mathbin{\mathsf{U}} (\dots (\boldsymbol{l}_n \mathbin{\mathsf{U}} \boldsymbol{r}_n) \dots)))), \tag{13}$$

where each $\boldsymbol{r}_i \in \mathcal{Q}^\sigma$ and each $\boldsymbol{l}_i$ is either in $\mathcal{Q}^\sigma$ or $\bot$ (recall that $\boldsymbol{q}$, $\boldsymbol{r}_i$, $\boldsymbol{l}_i$ have a single answer domain variable x and that we evaluate $\boldsymbol{q}$ at time point 0). Let $\mathcal{O}$ be an ontology. Then the class $\mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{Q}^\sigma)$ comprises *$\mathcal{O}$-peerless queries* in $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{Q}^\sigma)$, in which $\boldsymbol{r}_i \not\models_\mathcal{O} \boldsymbol{l}_i$ and $\boldsymbol{l}_i \not\models_\mathcal{O} \boldsymbol{r}_i$, for all $i \le n$. In what follows we write $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}$ instead of $\mathcal{O}, \mathcal{D}, 0, a \models \boldsymbol{q}$ when a is clear from context. We also write $\mathcal{D} \models \boldsymbol{q}$ instead of $\emptyset, \mathcal{D} \models \boldsymbol{q}$ (that is, for the empty ontology).

We illustrate our approach to characterising $\mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{Q}^\sigma)$-queries by a simple example.

Example 7. *Suppose $\boldsymbol{q} = \bigcirc A$ and $\sigma = \{A, B\}$. Then $E = (E^+, E^-)$ with $E^+ = \{\emptyset\{A\}\}$ and $E^- = \{\emptyset\{B\}\{A\}\}$ uniquely characterises $\boldsymbol{q}$ within the class $\mathit{LTL}^{\mathsf{U}}(\mathcal{P}^\sigma)$. Indeed, if $\boldsymbol{q}' = \boldsymbol{r}_0 \wedge (\boldsymbol{l}_1 \mathbin{\mathsf{U}} (\boldsymbol{r}_1 \wedge (\boldsymbol{l}_2 \mathbin{\mathsf{U}} (\dots (\boldsymbol{l}_n \mathbin{\mathsf{U}} \boldsymbol{r}_n) \dots)))) \in \mathit{LTL}_p^{\mathsf{U}}(\mathcal{P}^\sigma)$ fits E, then $\boldsymbol{q}' \equiv \boldsymbol{l}_1 \mathbin{\mathsf{U}} A$, for some $\boldsymbol{l}_1$. Because of E^-, $\boldsymbol{l}_1 \neq B$. So $\boldsymbol{l}_1 = \bot$ or $\boldsymbol{l}_1 = A$. In either of these cases, $\boldsymbol{q}' \equiv \bigcirc A$.*

A 'systematic way' of constructing E^- for $\boldsymbol{q} = \bigcirc A$ would be to include in it every $\emptyset\boldsymbol{l}\{A\}$ with $(\boldsymbol{l}, a) \in \mathcal{S}(\{A\}) = \{(\{B(a)\}, a)\}$. On the other hand, we cannot use the set $\{(\hat{\mathbf{m}}, a) \mid \mathbf{m} \in \mathcal{F}_{A(x)}\} = \{(\top, a)\}$ in place of $\mathcal{S}(\{A\})$. Intuitively, this is because the set of $\{\boldsymbol{l}_i \mid \emptyset\boldsymbol{l}_i\{A\} \in E^-\}$ must be such that, for every $\boldsymbol{l}$ in signature σ with $\boldsymbol{l} \not\models A$, we must have that some $\boldsymbol{l}_i \models \boldsymbol{l}$. However, the frontier $\mathcal{F}_A$ guarantees the above only for every $\boldsymbol{l}$ satisfying $A \models \boldsymbol{l}$.

In this section, we aim to prove the following general theorem:

Theorem 17. *Suppose $\mathcal{Q}$ is a class of domain queries, σ a signature, $\mathcal{L}$ has general split-partners within $\mathcal{Q}^\sigma$, and $\mathcal{O}$ is a σ-ontology in $\mathcal{L}$ that admits containment reduction. Then every query $\boldsymbol{q} \in \mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{Q}^\sigma)$ is uniquely characterisable wrt $\mathcal{O}$ within $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{Q}^\sigma)$. If a split-partner for any set Q of $\mathcal{Q}^\sigma$ queries wrt $\mathcal{O}$ within $\mathcal{Q}^\sigma$ is exponential, then there is an exponential-size unique characterisation of $\boldsymbol{q}$ wrt $\mathcal{O}$. If a split-partner of any set Q as above is polynomial and a split-partner $\mathcal{S}_\bot$ of $\bot(x)$ within $\mathcal{Q}^\sigma$ wrt $\mathcal{O}$ is a singleton, then there is a polynomial-size unique characterisation of $\boldsymbol{q}$ wrt $\mathcal{O}$.*

The proof is by reduction to the propositional case without ontology. We remind the reader of the result proved in [22]:

Theorem 18 ([22]). *$\mathit{LTL}_{pp}^{\mathsf{U}}(\mathcal{P}^\sigma)$ is polynomially characterisable within $\mathit{LTL}_p^{\mathsf{U}}(\mathcal{P}^\sigma)$ wrt the empty ontology with the characterisation given below*:

Let $\mathbf{s} \in \mathcal{P}^\sigma \cup \{\bot\}$*. We treat each such* $\mathbf{s} \neq \bot$ *as a set of its conjuncts and define* $\bar{\mathbf{s}} = \{A(a) \mid A(x) \in \mathbf{s}\}$*. For* $\mathbf{s} = \bot$*, we set* $\bar{\mathbf{s}} = \varepsilon$*, where* ε *is the empty word in the sense that* $\varepsilon\mathcal{D} = \mathcal{D}$*, for any data instance* $\mathcal{D}$*, and* $\varepsilon\varepsilon = \varepsilon$*. Consider* $\boldsymbol{q} \in \mathit{LTL}^{\mathsf{U}}_{pp}(\mathcal{P}^\sigma)$ *of the form* (13)*. Then* $\boldsymbol{q}$ *is uniquely characterised within* $\mathit{LTL}^{\mathsf{U}}_{p}(\mathcal{P}^\sigma)$ *by the example set* $E = (E^+, E^-)$*, where* E^+ *contains all data instances of the following forms*:

($\mathfrak{p}_0$) $\bar{\boldsymbol{r}}_0 \dots \bar{\boldsymbol{r}}_n$,

($\mathfrak{p}_1$) $\bar{\boldsymbol{r}}_0 \dots \bar{\boldsymbol{r}}_{i-1}\bar{\boldsymbol{l}}_i\bar{\boldsymbol{r}}_i \dots \bar{\boldsymbol{r}}_n = \mathcal{D}^i_{\boldsymbol{q}}$,

($\mathfrak{p}_2$) $\bar{\boldsymbol{r}}_0 \dots \bar{\boldsymbol{r}}_{i-1}\bar{\boldsymbol{l}}^k_i\bar{\boldsymbol{r}}_i \dots \bar{\boldsymbol{r}}_{j-1}\bar{\boldsymbol{l}}_j\bar{\boldsymbol{r}}_j \dots \bar{\boldsymbol{r}}_n = \mathcal{D}^j_{i,k}$*, for* $i < j$*, and* $k = 1, 2$;

and E^- *contains all instances* $\mathcal{D}$ *with* $\mathcal{D} \not\models \boldsymbol{q}$ *of the forms*:

($\mathfrak{n}_0$) $\bar{\sigma}^n$ *and* $\bar{\sigma}^{n-i}\overline{\sigma \setminus \{A\}}\bar{\sigma}^i$*, for* $A(x) \in \boldsymbol{r}_i$,

($\mathfrak{n}_1$) $\bar{\boldsymbol{r}}_0 \dots \bar{\boldsymbol{r}}_{i-1}\overline{\sigma \setminus X}\bar{\boldsymbol{r}}_i \dots \bar{\boldsymbol{r}}_n$*, for* $X = \{A, B\}$ *with* $A(x) \in \boldsymbol{l}_i$*,* $B(x) \in \boldsymbol{r}_i$*,* $X = \emptyset$*, and* $X = \{A\}$ *with* $A(x) \in \boldsymbol{l}_i$,

($\mathfrak{n}_2$) *for all* i *and* $A(x) \in \boldsymbol{l}_i \cup \{\bot(x)\}$, some *data instance*

$$\mathcal{D}^i_A = \bar{\boldsymbol{r}}_0 \dots \bar{\boldsymbol{r}}_{i-1}(\overline{\sigma \setminus \{A\}})\bar{\boldsymbol{r}}_i\bar{\boldsymbol{l}}^{k_{i+1}}_{i+1} \dots \bar{\boldsymbol{l}}^{k_n}_n\bar{\boldsymbol{r}}_n, \tag{14}$$

if any, such that $\max(\mathcal{D}^i_A) \leq (n+1)^2$ *and* $\mathcal{D}^i_A \not\models \boldsymbol{q}^\dagger$ *for* $\boldsymbol{q}^\dagger$ *obtained from* $\boldsymbol{q}$ *by replacing* $\boldsymbol{l}_j$*, for all* $j \leq i$*, with* $\bot$*. Note that* $\mathcal{D}^i_A \not\models \boldsymbol{q}$ *for peerless* $\boldsymbol{q}$*.*

Returning to proof of Theorem 17, assume a signature σ, an ontology $\mathcal{O}$ in σ, and a $\boldsymbol{q} \in \mathit{LTL}^{\mathsf{U}}_{pp}(\mathcal{Q}^\sigma)$ of the form (13) are given. We may assume that $\boldsymbol{r}_n \not\equiv_{\mathcal{O}} \top$. We obtain the set E^+ of positive examples as

($\mathfrak{p}'_0$) $\hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_n$,

($\mathfrak{p}'_1$) $\hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1}\hat{\boldsymbol{l}}_{i\_}i \dots \hat{\boldsymbol{r}}_n = \mathcal{D}^i_{\boldsymbol{q}}$,

($\mathfrak{p}_2$)$'$ $\hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1}\hat{\boldsymbol{l}}^k_i\hat{\boldsymbol{r}}_i \dots \hat{\boldsymbol{r}}_{j-1}\hat{\boldsymbol{l}}_j\hat{\boldsymbol{r}}_j \dots \hat{\boldsymbol{r}}_n = \mathcal{D}^j_{i,k}$, for $i < j$ and $k = 1, 2$.

We obtain the set E^- of negative examples by taking the following data instances $\mathcal{D}$ whenever $\mathcal{D} \not\models \boldsymbol{q}$:

($\mathfrak{n}'_0$) $\mathcal{A}_1, \dots, \mathcal{A}_n$ and $\mathcal{A}_1, \dots, \mathcal{A}_{n-i}, \mathcal{A}, \mathcal{A}_{n-i+1}, \dots, \mathcal{A}_n$, for $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{r}_i\})$ and $(\mathcal{A}_1, a), \dots, (\mathcal{A}_n, a) \in \mathcal{S}_\bot$;
($\mathfrak{n}'_1$) $\mathcal{D} = \hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1}\mathcal{A}\hat{\boldsymbol{r}}_i \dots \hat{\boldsymbol{r}}_n$, where $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i, \boldsymbol{r}_i\}) \cup \mathcal{S}(\{\boldsymbol{l}_i\}) \cup \mathcal{S}_\bot$;
($\mathfrak{n}'_2$) for all i and $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i, \boldsymbol{r}_i\}) \cup \mathcal{S}(\{\boldsymbol{l}_i\}) \cup \mathcal{S}_\bot$, *some* data instance

$$\mathcal{D}^i_{\mathcal{A}} = \hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1}\mathcal{A}\hat{\boldsymbol{r}}_i\hat{\boldsymbol{l}}^{k_{i+1}}_{i+1}\hat{\boldsymbol{r}}_{i+1} \dots \hat{\boldsymbol{l}}^{k_n}_n\boldsymbol{r}_n,$$

if any, such that $\max(\mathcal{D}^i_{\mathcal{A}}) \leq (n+1)^2$ and $\mathcal{D}^i_{\mathcal{A}} \not\models \boldsymbol{q}^\dagger$ for $\boldsymbol{q}^\dagger$ obtained from $\boldsymbol{q}$ by replacing all $\boldsymbol{l}_j$, for $j \leq i$, with $\bot$.

We show that $\boldsymbol{q}$ is uniquely characterised by the example set $E = (E^+, E^-)$ wrt $\mathcal{O}$ within $\mathit{LTL}^{\mathsf{U}}_p(\mathcal{Q}^\sigma)$. Consider any query

$$\boldsymbol{q}' = \boldsymbol{r}'_0 \wedge (\boldsymbol{l}'_1 \mathbin{\mathsf{U}} (\boldsymbol{r}'_1 \wedge (\boldsymbol{l}'_2 \mathbin{\mathsf{U}} (\dots (\boldsymbol{l}'_m \mathbin{\mathsf{U}} \boldsymbol{r}'_m) \dots))))$$

in $\mathit{LTL}^{\mathsf{U}}_p(\mathcal{Q}^\sigma)$ such that $\boldsymbol{q}' \not\equiv_{\mathcal{O}} \boldsymbol{q}$. We can again assume that $\boldsymbol{r}'_m \not\equiv_{\mathcal{O}} \top$. Thus, in what follows we can safely ignore what the ontology $\mathcal{O}$ entails after the timepoint $\max\mathcal{D}$ for any database $\mathcal{D}$ as these points do not contribute to entailment of $\boldsymbol{q}$ or $\boldsymbol{q}'$. In order to show that $\boldsymbol{q}$ fits E and $\boldsymbol{q}'$ does not fit E, we need a few definitions.

We define a map f that reduces the 2D case to the 1D case. Consider the alphabet

$$\Gamma = \{\boldsymbol{r}_0, \dots, \boldsymbol{r}_n, \boldsymbol{l}_1, \dots, \boldsymbol{l}_n, \boldsymbol{r}'_0, \dots, \boldsymbol{r}_m, \boldsymbol{l}'_1, \dots, \boldsymbol{l}'_m, \} \setminus \{\bot\},$$

in which we regard the CQs $\boldsymbol{r}_i, \boldsymbol{l}_i, \boldsymbol{r}'_j, \boldsymbol{l}'_j$ as symbols. Let $\hat{\Gamma} = \{(\hat{\boldsymbol{a}}, a) \mid \boldsymbol{a} \in \Gamma\}$, that is, $\hat{\Gamma}$ consists of the pointed databases corresponding to the CQs $\boldsymbol{a} \in \Gamma$. For any CQ $\boldsymbol{a}$, we set

$$f(\boldsymbol{a}) = \{\boldsymbol{b}(x) \mid \boldsymbol{b} \in \Gamma \text{ and } \mathcal{O}, \hat{\boldsymbol{a}} \models \boldsymbol{b}(a)\}.$$

Similarly, for any pointed data instance $(\mathcal{A}, a)$, we set

$$f(\mathcal{A}, a) = \{\boldsymbol{b}(x) \mid \boldsymbol{b} \in \Gamma \text{ and } \mathcal{O}, \mathcal{A} \models \boldsymbol{b}(a)\}$$

and, for any temporal data instance $\mathcal{D} = \mathcal{A}_0, \dots, \mathcal{A}_k$ with point a, set

$$f(\mathcal{D}, a) = (f(\mathcal{A}_0, a), \dots, f(\mathcal{A}_k, a)),$$

which is a temporal data instance over the signature Γ. Finally, we define an $LTL_p^{\mathsf{U}}(\mathcal{P}^\Gamma)$-query

$$f(\boldsymbol{q}) = \rho_0 \wedge (\lambda_1 \mathbin{\mathsf{U}} (\rho_1 \wedge (\lambda_2 \mathbin{\mathsf{U}} (\dots (\lambda_n \mathbin{\mathsf{U}} \rho_n) \dots))))$$

by taking $\rho_i = f(\boldsymbol{r}_i)$ and $\lambda_i = f(\boldsymbol{l}_i)$, and similarly for $\boldsymbol{q}'$. It follows immediately from the definition that, for any data instance $\mathcal{D}$, we have $\mathcal{O}, \mathcal{D} \models \boldsymbol{q}$ iff $f(\mathcal{D}, a) \models f(\boldsymbol{q})$ and $\mathcal{O}, \mathcal{D} \models f(\boldsymbol{q}')$ iff $f(\mathcal{D}, a) \models f(\boldsymbol{q}')$.

We first observe that $f(\boldsymbol{q})$ is a peerless $LTL_p^{\mathsf{U}}(\mathcal{P}^\Gamma)$-query: indeed, since $\mathcal{O}, \hat{\boldsymbol{r}}_i \not\models \boldsymbol{l}_i(a)$, we have $\boldsymbol{l}_i \in f(\boldsymbol{l}_i) \setminus f(\boldsymbol{r}_i)$, and since $\mathcal{O}, \hat{\boldsymbol{l}}_i \not\models \boldsymbol{r}_i(a)$, we have $\boldsymbol{r}_i \in f(\boldsymbol{r}_i) \setminus f(\boldsymbol{l}_i)$. It follows that $f(\boldsymbol{q}) \not\equiv f(\boldsymbol{q}')$

Let $E_{\text{prop}} = (E^+_{\text{prop}}, E^-_{\text{prop}})$ be the example set defined for $f(\boldsymbol{q})$ using $(\mathfrak{p}_0)$–$(\mathfrak{p}_2)$ and $(\mathfrak{n}_0)$–$(\mathfrak{n}_2)$. By Theorem 18, $f(\boldsymbol{q})$ fits E_{prop} and $f(\boldsymbol{q}')$ does not fit E_{prop}.

A *satisfying root $\mathcal{O}$-homomorphism* for any query

$$\boldsymbol{r}_0 \wedge (\boldsymbol{l}_1 \mathbin{\mathsf{U}} (\boldsymbol{r}_1 \wedge (\boldsymbol{l}_2 \mathbin{\mathsf{U}} (\dots (\boldsymbol{l}_n \mathbin{\mathsf{U}} \boldsymbol{r}_n) \dots))))$$

in $\mathcal{D}, a = (\mathcal{A}_0, a) \dots, (\mathcal{A}_k, a)$ is a map h from $\{0, \dots, n\}$ to $\mathbb{N}$ such that $h(0) = 0$ and $h(i) < h(i+1)$ for $i < n$ and

- $\mathcal{O}, \mathcal{A}_{f(i)} \models \boldsymbol{r}_i(a)$;
- $\mathcal{O}, \mathcal{A}_{i'} \models \boldsymbol{l}_i(a)$ for all $i' \in (f(i), f(i+1))$.

Clearly, such a root $\mathcal{O}$-homomorphism exists iff the query is satisfied in $\mathcal{D}, a$. If the query is in $LTL_p^{\mathsf{U}}(\mathcal{P}^\sigma)$ and $\mathcal{O}$ is empty, then we call the homomorphism above a *satisfying homomorphism*.

We are now in a position to show that $\boldsymbol{q}$ fits E but $\boldsymbol{q}'$ does not fit E. It is immediate from the definitions that $\boldsymbol{q}$ fits E. So we show that $\boldsymbol{q}'$ does not fit E.

Assume first that $f(\boldsymbol{q}')$ is not entailed by some example in E^+_p. Then $\boldsymbol{q}'$ is not entailed by some example in E^+ as the examples from $(\mathfrak{p}_0)$–$(\mathfrak{p}_2)$ are exactly the f-images of the examples $(\mathfrak{p}'_0)$–$(\mathfrak{p}'_2)$.

Assume now that $f(\boldsymbol{q}')$ is entailed by all data instances in E^+_{prop} and is also entailed by some $\mathcal{D}$ from E^-_{prop}. We show that then there is a data instance in E^- that entails $\boldsymbol{q}'$ under $\mathcal{O}$.

If $\mathcal{D} = \Gamma^n$, then it follows that the temporal depth of $f(\boldsymbol{q}')$ is less than the temporal depth of $f(\boldsymbol{q})$. Then $m < n$ and the query $\boldsymbol{q}'$ is entailed by some $\mathcal{A}_1, \dots, \mathcal{A}_n \in E^-$ with $(\mathcal{A}_i, a) \in \mathcal{S}_\bot$: we obtain $\mathcal{A}_i$ by taking $(\mathcal{A}_i, a) \in \mathcal{S}_\bot$ such that $\mathcal{O}, \mathcal{A}_i \models \boldsymbol{r}'_i(a)$.

Suppose $\mathcal{D} = \Gamma^{n-i}(\Gamma \setminus \{\boldsymbol{a}\})\Gamma^i \models f(\boldsymbol{q}')$. Observe that the only satisfying homomorphim that witnesses this is the identity mapping. So we have $f(\boldsymbol{r}_{n-i}) \not\subseteq \Gamma \setminus \{\boldsymbol{a}\}$ and therefore $\mathcal{O}, \hat{\boldsymbol{r}}_{n-i} \models \boldsymbol{a}(a)$ but $f(\boldsymbol{r}'_{n-i}) \subseteq \Gamma \setminus \{\boldsymbol{a}\}$. Then $\mathcal{O}, \hat{\boldsymbol{r}}'_{n-i} \not\models \boldsymbol{a}(a)$, and so $\boldsymbol{r}'_{n-i} \not\models_{\mathcal{O}} \boldsymbol{r}_i$. Therefore, there is $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{r}_{n-i}\})$ such that $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}'_{n-i}(a)$. Observe that also $\mathcal{O}, \mathcal{A} \not\models \boldsymbol{r}_{n-i}(a)$.

Now take, for any $j \neq n - i$, some $(\mathcal{A}_j, a) \in \mathcal{S}_\bot$ with $\mathcal{O}, \mathcal{A}_j \models \boldsymbol{r}'_j$. Then

$$\mathcal{O}, \mathcal{A}_0 \cdots \mathcal{A}_{n-i-1}\mathcal{A}\mathcal{A}_{n-i+1} \cdots \mathcal{A}_n \not\models \boldsymbol{q} \quad \text{and} \quad \mathcal{O}, \mathcal{A}_0 \cdots \mathcal{A}_{n-i-1}\mathcal{A}\mathcal{A}_{n-i+1} \cdots \mathcal{A}_n \not\models \boldsymbol{q}'.$$

Assume next that $\mathcal{D}$ is from $(\mathfrak{n}_1)$. We have $\mathcal{D} \models f(\boldsymbol{q}')$ and $\mathcal{D} \not\models f(\boldsymbol{q})$. As $\mathcal{D} \models f(\boldsymbol{q}')$, we have a satisfying homomorphism h for $f(\boldsymbol{q}')$ in $\mathcal{D}$.

If there is j such that $h(j) = i$, then let $\boldsymbol{r} = \boldsymbol{r}_j$. Otherwise, there is j such that $h(j) < i < h(j+1)$. Then let $\boldsymbol{r} = \boldsymbol{l}_j$. In both cases $f(\boldsymbol{r}) \subseteq Y$, where Y depends on $\mathcal{D}$ and is either:

1. $\Gamma \setminus \{\boldsymbol{a}, \boldsymbol{b}\}$ with $\mathcal{O}, \hat{\boldsymbol{l}}_i \models \boldsymbol{a}(a)$ and $\mathcal{O}, \hat{\boldsymbol{r}}_i \models \boldsymbol{b}(a)$ or
2. $\Gamma \setminus \{\boldsymbol{a}\}$ with $\mathcal{O}, \hat{\boldsymbol{l}}_i \models \boldsymbol{a}(a)$ or
3. Γ (only if $\boldsymbol{l}_i = \bot$) or
4. $\Gamma \setminus \{\boldsymbol{b}\}$.

Case 1. We have $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{a}(a)$ and $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{b}(a)$. Hence $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{l}_i(a)$ and $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{r}_i(a)$. By the definition of split-partners, there exists $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i, \boldsymbol{r}_i\})$ such that $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}(a)$. But then h is also a satisfying root $\mathcal{O}$-homomorphism in $\mathcal{D}', a$ witnessing that $\boldsymbol{q}'$ is entailed by $\mathcal{D}' = \hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1}\mathcal{A}\hat{\boldsymbol{r}}_i\hat{\boldsymbol{r}}_{i+1} \dots \boldsymbol{r}_n$ wrt $\mathcal{O}$.

It remains to show that $\mathcal{O}, \mathcal{D}' \not\models \boldsymbol{q}$. Assume otherwise. Take a satisfying root $\mathcal{O}$-homomorphism h^* witnessing $\mathcal{O}, \mathcal{D}' \models \boldsymbol{q}$. By peerlessness of $\boldsymbol{q}$, $h^*(j) = j$ for all $j < i$. But then $\mathcal{O}, \mathcal{A} \models \boldsymbol{l}_i$ or $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}_i$ which both contradict to $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i, \boldsymbol{r}_i\})$.

Case 2. We have $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{a}(a)$. Hence $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{l}_i(a)$. We now distinguish two cases. If also $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{r}_i(a)$, then we proceed as in the previous case and choose a split-partner $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i, \boldsymbol{r}_i\})$ such that $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}(a)$. We proceed as in Case 1.

If $\mathcal{O}, \hat{\boldsymbol{r}} \models \boldsymbol{r}_i(a)$, then we proceed as follows. Choose a split-partner $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i\})$ such that $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}(a)$. Then h is also a satisfying root $\mathcal{O}$-homomorphism in $\mathcal{D}', a$ witnessing that $\boldsymbol{q}'$ is entailed by $\mathcal{D}' = \hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1} \mathcal{A} \hat{\boldsymbol{r}}_i \hat{\boldsymbol{r}}_{i+1} \dots \boldsymbol{r}_n$ wrt $\mathcal{O}$.

It remains to show that $\mathcal{O}, \mathcal{D}' \not\models \boldsymbol{q}$. Assume otherwise. Take a satisfying root $\mathcal{O}$-homomorphism h^* in $\mathcal{D}', a$ witnessing $\mathcal{D}' \models \boldsymbol{q}$. By peerlessness of $\boldsymbol{q}$, $h^*(j) = j$ for all $j < i$. Then $h^*(i) = i$ as $\mathcal{O}, \mathcal{A} \models \boldsymbol{l}_i$ would contradict $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i\})$. But then h^* is a satisfying homomorphism in $\mathcal{D}, a$ witnessing $\mathcal{D} \models f(\boldsymbol{q})$ and we have derived a contradiction.

Case 3. We set $\mathcal{D}' = \hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1} \mathcal{A} \hat{\boldsymbol{r}}_i \hat{\boldsymbol{l}}_{i+1}^{k_{i+1}} \hat{\boldsymbol{r}}_{i+1} \dots \hat{\boldsymbol{l}}_n^{k_n} \boldsymbol{r}_n$ for some $(\mathcal{A}, a) \in \mathcal{S}_\perp$ with $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}'_i$. It directly follows from $\mathcal{D} \models f(\boldsymbol{q}')$ that $\mathcal{O}, \mathcal{D}' \models \boldsymbol{q}'$ and also from $\mathcal{D} \not\models f(\boldsymbol{q})$ that $\mathcal{O}, \mathcal{D}' \not\models \boldsymbol{q}$.

Case 4. We have $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{b}(a)$. Hence $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{r}_i(a)$. We distinguish two cases. If also $\mathcal{O}, \hat{\boldsymbol{r}} \not\models \boldsymbol{l}_i(a)$, then we proceed as in Case 1 and choose split-partner $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i, \boldsymbol{r}_i\})$ such that $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}(a)$.

If $\mathcal{O}, \hat{\boldsymbol{r}} \models \boldsymbol{l}_i(a)$, then we proceed as follows. Choose a split-partner $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{r}_i\})$ such that $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}(a)$. Then h is also a satisfying root $\mathcal{O}$-homomorphism in $\mathcal{D}', a$ witnessing that $\boldsymbol{q}'$ is entailed by $\mathcal{D}' = \hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1} \mathcal{A} \hat{\boldsymbol{r}}_i \hat{\boldsymbol{r}}_{i+1} \dots \boldsymbol{r}_n$ wrt $\mathcal{O}$.

It remains to show that $\mathcal{O}, \mathcal{D}' \not\models \boldsymbol{q}$. Assume otherwise. Take a satisfying root $\mathcal{O}$-homomorphism h^* in $\mathcal{D}', a$ witnessing $\mathcal{O}, \mathcal{D}' \models \boldsymbol{q}$. By peerlessness of $\boldsymbol{q}$, $h^*(j) = j$ for all $j < i$. Then $h^*(i) > i$ as $\mathcal{O}, \mathcal{A} \models \boldsymbol{r}_i(a)$ would contradict the definition of $\mathcal{A}$. But then, as $\Gamma \setminus \{\boldsymbol{b}\} \models f(\boldsymbol{l}_i)$, h^* is also a satisfying homomorphism in $\mathcal{D}, a$ witnessing $\mathcal{D} \models f(\boldsymbol{q})$ and we have derived a contradiction.

The case when $\mathcal{D}$ is from $(\mathfrak{n}_2)$ is considered similarly to $(\mathfrak{n}_1)$.

As a consequence of Theorems 17, 10, and [22, Theorem 30], we obtain:

Theorem 19. (i) *Every* $\boldsymbol{q} \in \mathit{LTL}_{pp}^{\mathsf{U}}(\mathit{ELIQ}^\sigma)$ *is exponentially uniquely characterisable wrt any horn$\mathcal{ALCHI}$ ontology in* σ *within* $\mathit{LTL}_p^{\mathsf{U}}(\mathit{ELIQ}^\sigma)$.

(ii) *Every* $\boldsymbol{q} \in \mathit{LTL}_{pp}^{\mathsf{U}}(\mathit{ELQ}^\sigma)$ *is polynomially uniquely characterisable wrt the empty ontology within* $\mathit{LTL}_p^{\mathsf{U}}(\mathit{ELQ}^\sigma)$.

6 Learning

We apply the results on unique characterisability to exact learnability of queries in $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}$(ELIQ) wrt ontologies. More precisely, given some class $\mathcal{Q}$ of two-dimensional queries and an ontology $\mathcal{O}$, we aim to identify a *target query* $\boldsymbol{q}_T \in \mathcal{Q}$ using queries to an oracle. It is assumed that $\boldsymbol{q}_T$ uses only symbols that occur in $\mathcal{O}$. A *positive example* (*for* $\boldsymbol{q}_T$ *wrt* $\mathcal{O}$) is a pointed temporal data instance $\mathcal{D}, a$ with $\mathcal{O}, \mathcal{D}, 0, a \models \boldsymbol{q}_T$. A *membership query* is then a pointed temporal data instance $\mathcal{D}, a$, and the (truthful) answer of the oracle is 'yes' if $\mathcal{D}, a$ is a positive example and 'no' otherwise. Let $\mathcal{L}$ be an ontology language. We call $\mathcal{Q}$ *polynomial-time learnable wrt $\mathcal{L}$-ontologies using membership queries* if there is a learning algorithm that receives an $\mathcal{L}$-ontology $\mathcal{O}$ and a positive example $\mathcal{D}, a$ with $\mathcal{D}$ satisfiable wrt $\mathcal{O}$, and constructs $\boldsymbol{q}_T$ (up to equivalence wrt $\mathcal{O}$) in time polynomial in the size of $\boldsymbol{q}_T, \mathcal{O}$, and $\mathcal{D}$. We also consider a weaker complexity measure that only bounds the number of queries but not the resources used between the queries: $\mathcal{Q}$ is called *polynomial query learnable wrt $\mathcal{L}$-ontologies using membership queries* if there is a learning algorithm that receives an ontology $\mathcal{O}$ and a positive example $\mathcal{D}, a$ with $\mathcal{D}$ satisfiable wrt $\mathcal{O}$, and constructs $\boldsymbol{q}_T$ (up to equivalence wrt $\mathcal{O}$) using polynomially-many queries of polynomial size in the size of $\boldsymbol{q}_T, \mathcal{O}$, and $\mathcal{D}$.

Before we can state the main technical result of this section, we introduce two further conditions on ontologies: polynomially bounded generalisations and polynomial-time instance checking. An ontology language $\mathcal{L}$ *admits polynomial-time instance checking* if given an $\mathcal{L}$-ontology $\mathcal{O}$, a labeled instance $\mathcal{D}, a$, and a concept name A, it is decidable in polynomial time whether $\mathcal{O}, \mathcal{D} \models A(a)$. Let $\boldsymbol{q}$ be a CQ. We say that $\boldsymbol{q}$ is $(\boldsymbol{q}_T, \mathcal{O})$*-minimal* if $\boldsymbol{q}' \not\models_{\mathcal{O}} \boldsymbol{q}_T$, for every restriction $\boldsymbol{q}'$ of $\boldsymbol{q}$ to a strict subset of the variables in $\boldsymbol{q}$. Moreover, $\boldsymbol{q}$ is $\mathcal{O}$*-saturated* if $\mathcal{O}, \hat{\boldsymbol{q}} \models A(y)$ implies that $A(y)$ is a conjunct in $\boldsymbol{q}$, for every variable y in $\boldsymbol{q}$ and every concept name A that occurs in $\mathcal{O}$. A *generalisation sequence for* $\boldsymbol{q}$ *wrt* $\mathcal{O}$ is a sequence $\boldsymbol{q}_1, \boldsymbol{q}_2, \dots$ of CQs that satisfies the following conditions, for all $i \geq 1$: $\boldsymbol{q}_i \models_{\mathcal{O}} \boldsymbol{q}_{i+1}$ and $\boldsymbol{q}_{i+1} \not\models_{\mathcal{O}} \boldsymbol{q}_i$, and $\boldsymbol{q}_i \models_{\mathcal{O}} \boldsymbol{q}$. We say that $\mathcal{O}$ has *polynomially-bounded generalisation* if for every CQ $\boldsymbol{q}$, any generalisation sequence $\boldsymbol{q}_1, \boldsymbol{q}_2, \dots$ towards $\boldsymbol{q}$ such that all $\boldsymbol{q}_i$ are satisfiable wrt $\mathcal{O}$, $(\boldsymbol{q}_T, \mathcal{O})$-minimal, and $\mathcal{O}$-saturated has length bounded by a polynomial in the size of $\boldsymbol{q}$ and $\mathcal{O}$.

Throughout the section, we consider ontologies in normal form. We strongly conjecture that this restriction to normal form is not needed, that is, all results below hold for ontologies that are not necessarily in normal form.[1] However, the normal form enables us to easily lift several results obtained for $\mathcal{ELIF}^\perp$ [25] to the two-dimensional setting. An $\mathcal{ELHIF}^\perp$-ontology is in *normal form* if every concept inclusion takes one of the following forms:

$$A \sqsubseteq \exists R.A', \qquad \exists R.A \sqsubseteq A', \qquad A \sqcap A' \sqsubseteq B,$$

where A, A' are concept names or $\top$, B is a concept name or $\perp$, and R is a role. It has been shown that $\mathcal{ELIF}^\perp$ ontologies in normal form have polynomially-bounded generalisations [25, Theorem 13], and using the same techniques it can be proved that the same is true for $\mathcal{ELHIF}^\perp$ ontologies.

Theorem 20. *Let $\mathcal{L}$ be an ontology language that contains only $\mathcal{ELHIF}^\perp$-onto-logies in normal form and that admits polysize frontiers within ELIQ that can be computed. Then:*

(i) *The class of safe queries in LTL$_p^{\bigcirc\Diamond\Diamond_r}$(ELIQ) is polynomial query learnable wrt $\mathcal{L}$-ontologies using membership queries.*

(ii) *The class LTL$_p^{\bigcirc\Diamond\Diamond_r}$(ELIQ) is polynomial query learnable wrt $\mathcal{L}$-ontologies using membership queries if the learner knows the temporal depth of the target query in advance.*

(iii) *The class LTL$_p^{\bigcirc\Diamond}$(ELIQ) is polynomial query learnable wrt $\mathcal{L}$-ontologies using membership queries.*

If $\mathcal{L}$ further admits polynomial-time instance checking and polynomial-time computable frontiers within ELIQ, then in (ii) and (iii), polynomial query learnability can be replaced by polynomial-time learnability. If, in addition, meet-reducibility of ELIQs wrt $\mathcal{L}$-ontologies can be decided in polynomial time, then also in (i) polynomial query learnability can be replaced by polynomial-time learnability.

Let $\mathcal{L}$ be an ontology language as in the theorem. We start with proving (i). Let $\boldsymbol{q}_T$ be the target query, $\mathcal{O}$ an $\mathcal{L}$-ontology, and $\mathcal{D}, a$ be a positive example with $\mathcal{D} = \mathcal{A}_0 \dots \mathcal{A}_n$ and $\mathcal{D}$ satisfiable wrt $\mathcal{O}$. We modify $\mathcal{D}$ in a number of steps such that in the end $\mathcal{D}$ viewed as a two-dimensional query is equivalent to $\boldsymbol{q}_T$. As a general proviso we assume that at all times each $\mathcal{A}_i$ (viewed as CQ) is $\mathcal{O}$-saturated; this is without loss of generality since instance checking wrt $\mathcal{ELHIF}^\perp$-ontologies is decidable [47]. By Lemma 5, we can assume $\boldsymbol{q}_T$ to be in normal form. We further assume $\boldsymbol{q}_T$ to be of shape (9):

$$\boldsymbol{q}_T = \boldsymbol{q}_0 \mathcal{R}_1 \boldsymbol{q}_1 \dots \mathcal{R}_n \boldsymbol{q}_n,$$

As $\boldsymbol{q}_T$ is safe, it does not have lone conjuncts.

Step 1. We first aim to find a temporal data instance which is *tree-shaped*, meaning that in $\mathcal{D} = \mathcal{A}_0 \dots \mathcal{A}_n$ each $\mathcal{A}_i$ is tree-shaped. To achieve this, we exhaustively apply the following rules Unwind and Minimise with a preference given to Minimise. A *cycle* in a data instance is a sequence $R_1(a_1, a_2), \dots, R_n(a_n, a_1)$ of distinct atoms such that $a_1, \dots, a_n$ are distinct.

Minimise. If there is some i and some individual $b \in \mathit{ind}(\mathcal{A}_i)$ such that $\mathcal{D}', a$ is a positive example where $\mathcal{D}'$ is obtained from $\mathcal{D}$ by dropping from $\mathcal{A}_i$ all atoms that mention b, then replace $\mathcal{D}$ with $\mathcal{D}'$.

Unwind. Choose an atom $R(a, b) \in \mathcal{A}_i$ that is part of a cycle. Obtain $\mathcal{A}_i'$ by first adding a disjoint copy $\mathcal{A}_i'$ of $\mathcal{A}_i$ to $\mathcal{A}_i$ and let a', b' be the copies of a, b in $\mathcal{A}_i'$. Then replace all atoms $S(a, b)$ (respectively, $S(a', b')$) by $S(a, b')$ (respectively, $S(a', b)$), for all roles S.

It is clear that the resulting temporal data instance is tree-shaped as required.

Step 2. Repeat the following until $\mathcal{D}$ stabilizes. If there is some i such that $\mathcal{D}', a$ is a positive example where $\mathcal{D}'$ is obtained from $\mathcal{D}$ by dropping $\mathcal{A}_i$ from $\mathcal{D}$, then replace $\mathcal{D}$ with $\mathcal{D}'$. Clearly, the length of the resulting temporal data sequence $\mathcal{D}$ is at least as large as the number of occurrences of suc and $<$ in $\boldsymbol{q}_T$. Finally, observe that every root $\mathcal{O}$-homomorphism $h : \boldsymbol{q}_T \to \mathcal{D}$ is block surjective for the block size $b := \max(\mathcal{D}) + 1$ (note that $\mathcal{D}$ has only one block for this b).

[1] More concretely, we conjecture that learnability of two-dimensional query classes $\mathcal{Q}$ wrt $\mathcal{ELHIF}^\perp$-ontologies can be reduced to learnability of $\mathcal{Q}$ wrt $\mathcal{ELHIF}^\perp$-ontologies in normal form just as in the non-temporal case; cf. [25, Lemma 10].

Step 3. In this step, we 'close' $\mathcal{D}$ under applications of the Rules (a)–(e) used in Lemma 6. Formally, consider the following Rule 3(x), for x $\in$ {a,b,c,d,e}.

3(x) Let $\mathcal{D}'$ be a data instance obtained from $\mathcal{D}$ by applying Rule (x). If $\mathcal{D}', a$ is a positive example, replace $\mathcal{D}$ with the result of the exhaustive application of Minimise to $\mathcal{D}'$.

We first apply 3(b) and 3(c) until $\mathcal{D}$ stabilises. Then, we exhaustively apply 3(a), 3(d), and 3(e) giving preference to 3(a).

After *Step 3*, $\mathcal{D}$ satisfies that, if $\mathcal{D}'$ is the result of an application of Rules (a)–(e), then $\mathcal{D}', a$ is not a positive example.

Step 4. In this step, we take care of lone conjuncts in $\mathcal{D}$ by applying $(*)$ below as long as $\mathcal{D}$ contains one. Recall that, by Lemma 3, a CQ $\boldsymbol{q}$ is meet-reducible iff $|\mathcal{F}_{\boldsymbol{q}}| \geq 2$ provided that $\boldsymbol{q}' \not\models_{\mathcal{O}} \boldsymbol{q}''$, for all distinct $\boldsymbol{q}', \boldsymbol{q}'' \in \mathcal{F}_{\boldsymbol{q}}$. We can compute such a minimal frontier $\mathcal{F}_{\boldsymbol{q}}$ by first computing any frontier F (which is possible by assumption) and then exhaustively removing from F queries $\boldsymbol{q}''$ such that $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}''$ for some $\boldsymbol{q}' \in F$ with $\boldsymbol{q}' \neq \boldsymbol{q}''$. Note that the test $\boldsymbol{q}' \models_{\mathcal{O}} \boldsymbol{q}''$ is decidable for $\mathcal{ELHIF}^{\perp}$-ontologies [14].

$(*)$ Choose a primitive block $\emptyset^b \mathcal{A} \emptyset^b$ in $\mathcal{D}$ such that $|\mathcal{F}_{\boldsymbol{q}}| \geq 2$, for the ELIQ $\boldsymbol{q}$ with $\mathcal{A} = \hat{\boldsymbol{q}}$. Let $\mathcal{F}_{\boldsymbol{q}} = \{\boldsymbol{q}_1, \dots, \boldsymbol{q}_\ell\}$ and $w = \hat{\boldsymbol{q}}_1 \emptyset^b \hat{\boldsymbol{q}}_2 \emptyset^b \dots \hat{\boldsymbol{q}}_\ell \emptyset^b$. Denote with $\mathcal{D}_k$ the result of replacing $\emptyset^b \mathcal{A} \emptyset^b$ in $\mathcal{D}$ with $\emptyset^b (w)^k$. Then identify some $i \geq 1$ such that $\mathcal{D}_i, a$ is a positive example, by using membership queries for $i = 1, 2, \dots$. Notice that this requires only polynomially-many membership queries as $\mathcal{D}_k, a$ is a positive example for $k = |\boldsymbol{q}_T|$, and that all queries are of polynomial size since $\mathcal{F}_{\boldsymbol{q}}$ is of polynomial size. Replace $\mathcal{D}$ with the result of exhaustively applying Rule 3(a) to $\mathcal{D}_i$ and subsequently shortening blocks $\emptyset^d$ for $d > b$ to $\emptyset^b$.

Let $\mathcal{D}$ be the result of *Step 4*. It is routine to verify that 3(a)–3(e) are not applicable, that $\mathcal{D}$ is b-normal, and that $\mathcal{D}$ is without lone conjuncts wrt $\mathcal{O}$ within $\mathit{LTL}_p^{\bigcirc\Diamond\Diamond_r}$(ELIQ). By Lemma 6, any root $\mathcal{O}$-homomorphism is a root $\mathcal{O}$-isomorphism. Thus, the algorithm has identified all blocks in the following sense. Suppose that $\boldsymbol{q}_T = \boldsymbol{q}_0 \mathcal{R}_1 \boldsymbol{q}_1 \dots \mathcal{R}_m \boldsymbol{q}_m$ is a sequence of blocks $\boldsymbol{q}_i = \boldsymbol{r}_0^i \dots \boldsymbol{r}_{\ell_i}^i$ and

$$\mathcal{D} = \mathcal{D}_0 \emptyset^b \mathcal{D}_1 \dots \emptyset^b \mathcal{D}_n \text{ where } \mathcal{D}_i = \mathcal{A}_0^i \dots \mathcal{A}_{k_i}^i.$$

Then $m = n$ and each block $\mathcal{D}_i$ in $\mathcal{D}$ is isomorphic to $\boldsymbol{q}_i$, that is, $\ell_i = k_i$ and $\hat{\boldsymbol{r}}_j^i = \mathcal{A}_j^i$, for all i, j with $0 \leq i \leq n$ and $0 \leq j \leq k_i$. It is unclear, however, whether the $\mathcal{R}_i$ are (a single) $\leq$ or a sequence of $<$. This is resolved in the final step.

Step 5. We determine $\mathcal{R}_{i+1}$, for each i with $0 \leq i < n$, as follows:

- If $\boldsymbol{r}_{k_i}^i$ and $\boldsymbol{r}_0^{i+1}$ are compatible wrt $\mathcal{O}$ and $\mathcal{D}_i, a$ with $\mathcal{D}_i = \mathcal{D}_0 \emptyset^b \dots \emptyset^b \mathcal{D}_i \bowtie \mathcal{D}_{i+1} \emptyset^b \dots \emptyset^b \mathcal{D}_n$ ($\bowtie$ defined as in the proof of Theorem 15) is a positive example, then $\mathcal{R}_{i+1}$ is $\leq$. Otherwise, let s be minimal such that $\mathcal{D}_i', a$ is a positive example for $\mathcal{D}_i' = \mathcal{D}_0 \emptyset^b \dots \emptyset^b \mathcal{D}_i \emptyset^s \mathcal{D}_{i+1} \emptyset^b \dots \emptyset^b \mathcal{D}_n$. Then, $\mathcal{R}_{i+1}$ is a sequence of s times $<$.

We show now that indeed the returned query is equivalent to $\boldsymbol{q}_T$ and that the algorithm issues only polynomially-many membership queries. We analyse *Steps 1–5* separately.

For *Step 1*, we lift the notion of generalisation sequences to temporal data instances. For the sake of convenience, we treat the data instances in the time points as CQs. A sequence $\mathcal{D}_1, \dots$ of temporal data instances is a *generalisation sequence towards $\boldsymbol{q}$ wrt $\mathcal{O}$* if for all $i \geq 1$:

- $\mathcal{D}_{i+1}$ is obtained from $\mathcal{D}_i$ by modifying one non-temporal CQ $\boldsymbol{r}_j$ in $\mathcal{D}_i$ to $\boldsymbol{r}_j'$ such that $\boldsymbol{r}_j \models_{\mathcal{O}} \boldsymbol{r}_j'$ and $\boldsymbol{r}_j' \not\models_{\mathcal{O}} \boldsymbol{r}_j$;
- $\mathcal{O}, \mathcal{D}_i \models \boldsymbol{q}$ for all $i \geq 1$.

Lemma 7. *Let $\boldsymbol{q}$ be a CQ. The length of a generalisation sequence $\mathcal{D}_1, \dots, \mathcal{D}_n$ towards $\boldsymbol{q}$ wrt $\mathcal{O}$ such that all $\mathcal{D}_i$ are satisfiable wrt $\mathcal{O}$, $\mathcal{O}$-saturated, and $(\boldsymbol{q}, \mathcal{O})$-minimal is bounded by a polynomial in the size of $\boldsymbol{q}$, $\mathcal{O}$, and $\mathcal{D}_1$.*

Proof. Consider a time point i and let $\boldsymbol{r}_1, \boldsymbol{r}_2, \dots$ be the sequence of different queries at time point i that occur in the generalisation sequence, that is, $\boldsymbol{r}_j \models_{\mathcal{O}} \boldsymbol{r}_{j+1}$ and $\boldsymbol{r}_{j+1} \not\models_{\mathcal{O}} \boldsymbol{r}_j$, for each j. Let h be a root homomorphism from $\boldsymbol{q}$ to $\mathcal{D}_n$ and let I be the set of all t with $h(t) = i$. (By construction, h is a root homomorphism from $\boldsymbol{q}$ to all $\mathcal{D}_j$.) Consider $\boldsymbol{q}' = \bigwedge_{i \in I} \boldsymbol{q}_i$. Clearly, $\boldsymbol{r}_1, \boldsymbol{r}_2, \dots,$ is a generalisation sequence towards $\boldsymbol{q}'$ wrt $\mathcal{O}$. Since all $\mathcal{D}_j$ are satisfiable wrt $\mathcal{O}$, $\mathcal{O}$-saturated and $(\boldsymbol{q}, \mathcal{O})$-minimal, it follows that in particular, all $\boldsymbol{r}_1, \boldsymbol{r}_2, \dots$ are satisfiable wrt $\mathcal{O}$, $\mathcal{O}$-saturated and $(\boldsymbol{q}, \mathcal{O})$-minimal. By [25, Theorem 13], the length of $\boldsymbol{r}_1, \boldsymbol{r}_2, \dots$ is bounded polynomially. Since there are only $\max(\mathcal{D}_1)$ time points to consider, the sequence $\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_n$ is polynomially bounded. ☐

Now, let $\mathcal{D}_1, \mathcal{D}_2, \dots$ be the sequence of temporal data instances that Unwind is applied to during *Step 1*. Clearly, all these queries are $(\boldsymbol{q}_T, \mathcal{O})$-minimal (recall that we give preference to Minimise) and $\mathcal{O}$-saturated. Since an application of Minimise decreases the overall number of individuals in the instance, there are only polynomially-many applications of Minimise between $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$. In the proof of Lemma 14 in [25], it is shown that the operation Unwind[2] applied to a $(\boldsymbol{q}_T, \mathcal{O})$-minimal CQ $\boldsymbol{q}$ leads to a strictly weaker CQ $\boldsymbol{q}'$, that is, $\boldsymbol{q} \models_{\mathcal{O}} \boldsymbol{q}'$, but not vice versa. This applies here as well, and implies that $\mathcal{D}_1, \mathcal{D}_2, \dots$ is a generalisation sequence towards $\boldsymbol{q}_T$ wrt $\mathcal{O}$. Applying Lemma 7 yields that *Step 1* terminates in time polynomial in the size of $\boldsymbol{q}_T$, $\mathcal{O}$, and $\mathcal{D}$.

It should be clear that only polynomially-many (in $\mathcal{D}$) membership queries are asked in *Step 2*.

We next analyse *Step 3*, starting with Rules 3(b) and 3(c). First note that the number of applications of Rules 3(b) and 3(c) is bounded by the number of $<$ and $\leq$ in $\boldsymbol{q}_T$. To see this, we inductively show that Rules 3(b) and 3(c) preserve the fact that every root $\mathcal{O}$-homomorphism $h : \boldsymbol{q}_T \to \mathcal{D}$ is block surjective. As noted above, this certainly holds before *Step 3*. Suppose now that $\mathcal{D}'$ is obtained by a single application of 3(b) or 3(c) to $\mathcal{D}$, and that there is a root $\mathcal{O}$-homomorphism that is not block surjective. Then we can easily construct a non-block surjective homomorphism from $\boldsymbol{q}_T$ to $\mathcal{D}$, a contradiction. Applications of Minimise also preserve the claim.

We next analyse Rules 3(a), 3(d), and 3(e). Let $\mathcal{D}_1, \mathcal{D}_2, \dots$ be a sequence of databases obtained by a sequence of applications of 3(a). Clearly, $\mathcal{D}_1, \mathcal{D}_2, \dots$ is a generalisation sequence towards $\boldsymbol{q}_T$ wrt $\mathcal{O}$ such that all $\mathcal{D}_i$ are satisfiable wrt $\mathcal{O}$, $\mathcal{O}$-saturated, and $(\boldsymbol{q}_T, \mathcal{O})$-minimal. By Lemma 7, the length of the sequence is polynomially bounded. Further note that applications of 3(a) preserve that every root $\mathcal{O}$-homomorphism is block surjective and that 3(b) and 3(c) remain not applicable.

Next consider an application of 3(d) to a temporal data instance $\mathcal{D}$ where 3(a) is not applicable and such that every root $\mathcal{O}$-homomorphism is block surjective. Let $\mathcal{D}'$ be the result. Since 3(a) is not applicable, every root $\mathcal{O}$-homomorphism to $\mathcal{D}'$ must also be block surjective. Thus, the number of applications of 3(d) is bounded by the number of $<$ and $\leq$ in $\boldsymbol{q}_T$. The same argument works for 3(e). It is readily seen that 3(b) and 3(c) are still not applicable, thus none of the rules is applicable to $\mathcal{D}$. Overall, we obtain a polynomial number of rule applications. This finishes the analysis of *Step 3*.

Consider now *Step 4*. As noted in $(*)$, identifying the right $\mathcal{D}_i$ needs only polynomially many membership queries (despite the fact that deciding meet-reducibility might require more time). Since exhaustive application of 3(a) requires only polynomial time, a single application of $(*)$ requires only polynomially many membership queries. Moreover, using the fact that 3(a) is not applicable before application of $(*)$ one can show that the number of 'gaps' is increased and that the rule preserves that every root $\mathcal{O}$-homomorphism is block surjective. Hence, $(*)$ is applied at most once for each $\leq$ in $\boldsymbol{q}_T$.

It remains to analyse the running time of *Step 5*. Clearly, only linearly many (in the size of $\boldsymbol{q}_T$) membership queries are asked. To finish the argument, we only note that $\mathcal{ELHIF}^{\bot}$ admits tractable containment reduction and that satisfiability wrt $\mathcal{ELHIF}^{\bot}$-ontologies is decidable.

We argue next that the above algorithm runs in polynomial time if $\mathcal{L}$ additionally admits polynomial-time instance checking, polynomial-time computable frontiers, and meet-reducibility of ELIQs wrt $\mathcal{L}$-ontologies can be decided in polynomial time. First note that $\mathcal{O}$-saturation of $\mathcal{D}$ (which is assumed throughout the algorithm) can be established in polynomial time via instance checking. Then observe that, in *Step 4*, a (not necessarily minimal) frontier $\mathcal{F}_{\boldsymbol{q}}$ can be computed in polynomial time and meet-reducibility can also be decided in polynomial time, by assumption. Together with the analysis of *Step 4* above, this yields that *Step 4* needs only polynomial time. Finally observe that also *Step 5* runs in polynomial time since the tests for satisfiability can be reduced to (polynomial time) instance checking.

It remains to prove Points (ii) and (iii) from Theorem 20. The learning algorithm for Point (ii) is similar to the algorithm provided above, but with a modified *Step 4* since in this case $\boldsymbol{q}_T$ might have lone conjuncts and possibly more than one variable from $var(\boldsymbol{q}_T)$ is mapped to the same time point in $\mathcal{D}$. Let T be the temporal depth of the target query.

Step 4′. In this step, we apply $(*')$ until $\mathcal{D}$ stabilises.

$(*')$ Choose a primitive block $\emptyset^b \mathcal{A} \emptyset^b$ and an ELIQ $\boldsymbol{q}$ with $\mathcal{A} = \hat{\boldsymbol{q}}$. Let $\mathcal{F}_{\boldsymbol{q}} = \{\boldsymbol{q}_1, \dots, \boldsymbol{q}_\ell\}$ and $w = \hat{\boldsymbol{q}}_1 \emptyset^b \hat{\boldsymbol{q}}_2 \emptyset^b \dots \hat{\boldsymbol{q}}_\ell \emptyset^b$. Let $\mathcal{D}'$ be the result of replacing $\emptyset^b \mathcal{A} \emptyset^b$ in $\mathcal{D}$ with $\emptyset^b (w)^T$. If $\mathcal{D}', a$ is a positive ex-

[2] Unwind is called *Double Cycle* in [25].

ample, then replace $\mathcal{D}$ with the result of exhaustively applying Rule 3(a) to $\mathcal{D}'$ and subsequently shortening blocks $\emptyset^d$ for $d > b$ to $\emptyset^b$.

Let $\mathcal{D}$ be the result of *Step 4*. It is routine to verify that 3(a)–3(e) are not applicable. The following can be proven similar to Lemma 6.

Lemma 8. *Let $\mathcal{O}$ and $\boldsymbol{q}_T$ be as above. Let b exceed the number of $\Diamond$ and $\bigcirc$ in $\boldsymbol{q}_T$, and let $\mathcal{D}$ be b-normal. If Rules 3(a)–(e) and $(*')$ are not applicable, then any root $\mathcal{O}$-homomorphim $h\colon \boldsymbol{q} \to \mathcal{D}$ is a root $\mathcal{O}$-isomorphism.*

As an immediate consequence, after *Step 4′* modified algorithm has identified all blocks in $\boldsymbol{q}_T$ as described above and it remains to apply *Step 5*.

The learning algorithm for Point (iii) is a similar modification. Note that for the query class *LTL*$_p^{\bigcirc\Diamond\Diamond_r}(ELIQ)$, after *Step 2*, we know *exactly* the temporal depth of $\boldsymbol{q}_T$, namely b. This b can then be used in place of T in $(*')$ above. This finishes the proof of Theorem 20. ☐

We apply Theorem 20 to concrete ontology languages, namely *DL-Lite*$_\mathcal{F}^-$ and *DL-Lite*$_\mathcal{H}$. Let us denote by *DL-Lite*$_\mathcal{F}^{-\text{nf}}$ and *DL-Lite*$_\mathcal{H}^{\text{nf}}$ the set of *DL-Lite*$_\mathcal{F}^-$ and *DL-Lite*$_\mathcal{H}$-ontologies in normal form, respectively.

Theorem 21. *The following learnability results hold:*

(i) *The class of safe queries in LTL*$_p^{\bigcirc\Diamond\Diamond_r}$*(ELIQ) is polynomial query learnable wrt DL-Lite*$_\mathcal{H}^{nf}$*-ontologies using membership queries and polynomial-time learnable wrt DL-Lite*$_\mathcal{F}^{-nf}$*-ontologies using membership queries.*

(ii) *The class LTL*$_p^{\bigcirc\Diamond\Diamond_r}$*(ELIQ) is polynomial-time learnable wrt both DL-Lite*$_\mathcal{F}^{-nf}$ *and DL-Lite*$_\mathcal{H}^{nf}$*-ontologies using membership queries if the learner knows the temporal depth of the target query in advance.*

(iii) *The class LTL*$_p^{\bigcirc\Diamond}$*(ELIQ) is polynomial query learnable wrt both DL-Lite*$_\mathcal{F}^{-nf}$ *and DL-Lite*$_\mathcal{H}^{nf}$*-ontologies using membership queries.*

Proof. The theorem is a direct consequence of Theorem 20 and the fact that the considered ontology languages satisfy all conditions mentioned in that theorem. Most importantly:

- *DL-Lite*$_\mathcal{F}$ admits polynomial-time instance checking [17] and *DL-Lite*$_\mathcal{F}^-$ admits polynomial-time computable frontiers [26]. Moreover, meet-reducibility in *DL-Lite*$_\mathcal{F}^-$ is decidable in polynomial time, by Lemma 4.
- *DL-Lite*$_\mathcal{H}$ admits polynomial-time instance checking [17] and admits polynomial-time computable frontiers [26].

This completes the proof. ☐

We repeat that we conjecture that Theorem 21 also holds for *DL-Lite*-ontologies not in normal form, but leave an investigation of this issue for future work. We also leave for future work whether or not *LTL*$_p^{\bigcirc\Diamond\Diamond_r}$(ELIQ) is even polynomial-*time* learnable wrt *DL-Lite*$_\mathcal{H}$-ontologies using membership queries. For this to be true, it suffices to show that meet-reducibility of $(\boldsymbol{q}_T, \mathcal{O})$-*minimal* ELIQs is decidable in polynomial time (but recall that deciding that for general ELIQs is NP-complete, by Lemma 4).

7 Outlook

Many interesting and challenging problems remain to be addressed. We discuss a few. (1) Is it possible to overcome some of our negative results for unique characterisability by admitting some form of infinite (but finitely presentable) examples? Some results in this direction without ontologies are obtained in [41]. (2) We focussed on temporal queries based on ELIQs but did not consider in any detail interesting subclasses such as the class ELQ of queries equivalent to $\mathcal{EL}$-concepts. We conjecture that in this case some new results on unique characterisability and learnability using membership queries can be obtained. (3) We did not consider learnability using membership queries of temporal queries with until. In fact, it remains completely open in how far our characterisability results for these queries can be exploited to obtain polynomial query (or time) learnability results. (4) We only considered queries in which no description logic operators are applied to temporal operator, and within this class only path queries. This is motivated by the negative results obtained in [22] where it is shown that (i) applying existential restrictions $\exists P$ to queries with $\Diamond$

and $\bigcirc$ quickly leads to non characterisability and (ii) that even without DL-operators and without ontology, branching queries with $\Diamond$ can often not be uniquely characterised. Still, there should be scope for useful positive characterisation results.

References

[1] Fides Aarts and Frits Vaandrager. Learning i/o automata. In *International Conference on Concurrency Theory*, pages 71–85. Springer, 2010.

[2] Bogdan Alexe, Balder ten Cate, Phokion G. Kolaitis, and Wang Chiew Tan. Characterizing schema mappings via data examples. *ACM Trans. Database Syst.*, 36(4):23, 2011.

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

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

[5] Marcelo Arenas and Gonzalo I. Diaz. The exact complexity of the first-order logic definability problem. *ACM Trans. Database Syst.*, 41(2):13:1–13:14, 2016.

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

[7] Alessandro Artale, Roman Kontchakov, Alisa Kovtunova, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev. Ontology-mediated query answering over temporal data: A survey. In *Proc. of TIME 2017*, volume 90 of *LIPIcs*, pages 1:1–1:37. Schloss Dagstuhl, Leibniz-Zentrum für Informatik, 2017.

[8] Alessandro Artale, Roman Kontchakov, Alisa Kovtunova, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev. First-order rewritability of ontology-mediated queries in linear temporal logic. *Artif. Intell.*, 299:103536, 2021.

[9] Alessandro Artale, Roman Kontchakov, Alisa Kovtunova, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev. First-order rewritability and complexity of two-dimensional temporal ontology-mediated queries. *J. Artif. Intell. Res.*, 75:1223–1291, 2022.

[10] Alessandro Artale, Roman Kontchakov, Vladislav Ryzhikov, and Michael Zakharyaschev. A cookbook for temporal conceptual data modelling with description logics. *ACM Trans. Comput. Log.*, 15(3):25:1–25:50, 2014.

[11] Franz Baader, Stefan Borgwardt, and Marcel Lippmann. Temporal query entailment in the description logic SHQ. *J. Web Semant.*, 33:71–93, 2015.

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

[13] Pablo Barceló and Miguel Romero. The complexity of reverse engineering problems for conjunctive queries. In *Proc. of ICDT*, pages 7:1–7:17, 2017.

[14] Meghyn Bienvenu, Peter Hansen, Carsten Lutz, and Frank Wolter. First order-rewritability and containment of conjunctive queries in Horn description logics. In *Proc. of IJCAI*, pages 965–971, 2016.

[15] Meghyn Bienvenu, Balder ten Cate, Carsten Lutz, and Frank Wolter. Ontology-based data access: A study through disjunctive datalog, CSP, and MMSNP. *ACM Trans. Database Syst.*, 39(4):33:1–33:44, 2014.

[16] Stefan Borgwardt and Veronika Thost. Temporal query answering in the description logic EL. In *Proc. of IJCAI 2015*, pages 2819–2825. AAAI Press, 2015.

[17] 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.

[18] Alberto Camacho and Sheila A. McIlraith. Learning interpretable models expressed in linear temporal logic. In *Proc. of ICAPS*, pages 621–630. AAAI Press, 2019.

[19] Sofia Cassel, Falk Howar, Bengt Jonsson, and Bernhard Steffen. Active learning for extended finite state machines. *Formal Aspects Comput.*, 28(2):233–263, 2016.

[20] C.C. Chang and H. Jerome Keisler. *Model Theory*. Elsevier, 1998.

[21] Nathanaël Fijalkow and Guillaume Lagarde. The complexity of learning linear temporal formulas from examples. *CoRR*, abs/2102.00876, 2021.

[22] Marie Fortin, Boris Konev, Vladislav Ryzhikov, Yury Savateev, Frank Wolter, and Michael Zakharyaschev. Unique characterisability and learnability of temporal instance queries. In *Proc. of KR*, 2022.

[23] Marie Fortin, Boris Konev, Vladislav Ryzhikov, Yury Savateev, Frank Wolter, and Michael Zakharyaschev. Reverse engineering of temporal queries mediated by LTL ontologies. In *Proceedings of IJCAI*, 2023.

[24] Maurice Funk, Jean Christoph Jung, and Carsten Lutz. Actively learning concepts and conjunctive queries under ELr-ontologies. In *Proc. of IJCAI 2021*, pages 1887–1893. ijcai.org, 2021.

[25] Maurice Funk, Jean Christoph Jung, and Carsten Lutz. Exact learning of ELI queries in the presence of dl-lite-horn ontologies. In *Proc. of DL*, volume 3263 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.

[26] Maurice Funk, Jean Christoph Jung, and Carsten Lutz. Frontiers and exact learning of ELI queries under dl-lite ontologies. In *Proc. of IJCAI*, pages 2627–2633, 2022.

[27] 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.

[28] Víctor Gutiérrez-Basulto, Jean Christoph Jung, and Roman Kontchakov. Temporalized EL ontologies for accessing temporal data: Complexity of atomic queries. In *Proc. of IJCAI*, pages 1102–1108. IJCAI/AAAI Press, 2016.

[29] 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*, 2018.

[30] Falk Howar and Bernhard Steffen. Active automata learning in practice - an annotated bibliography of the years 2011 to 2016. In *Machine Learning for Dynamic Software Analysis: Potentials and Limits, International Dagstuhl Seminar 16172*, volume 11026 of *Lecture Notes in Computer Science*, pages 123–148. Springer, 2018.

[31] Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. Logical separability of labeled data examples under ontologies. *Artif. Intell.*, 313:103785, 2022.

[32] Stanislav Kikot, Roman Kontchakov, and Michael Zakharyaschev. On (in)tractability of OBDA with OWL 2 QL. In Riccardo Rosati, Sebastian Rudolph, and Michael Zakharyaschev, editors, *Proc. of DL*. CEUR-WS.org, 2011.

[33] Phokion G. Kolaitis. Schema Mappings and Data Examples: Deriving Syntax from Semantics (Invited Talk). In *Proc. of FSTTCS*, volume 13, pages 25–25, Dagstuhl, Germany, 2011. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[34] Boris Konev, Carsten Lutz, Ana Ozaki, and Frank Wolter. Exact learning of lightweight description logic ontologies. *J. Mach. Learn. Res.*, 18:201:1–201:63, 2017.

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

[36] Caroline Lemieux, Dennis Park, and Ivan Beschastnikh. General ltl specification mining (t). In *Proc. of ASE*, pages 81–92. IEEE, 2015.

[37] Denis Mayr Lima Martins. Reverse engineering database queries from examples: State-of-the-art, challenges, and research opportunities. *Inf. Syst.*, 83:89–100, 2019.

[38] Ralph McKenzie. Equational bases and nonmodular lattice varieties. *Transactions of the American Mathematical Society*, 174:1–43, 1972.

[39] Daniel Neider and Ivan Gavran. Learning linear temporal properties. In *Proc. of FMCAD*, pages 1–10. IEEE, 2018.

[40] Ana Ozaki. Learning description logic ontologies: Five approaches. where do they stand? *Künstliche Intell.*, 34(3):317–327, 2020.

[41] Patric Sestic. Unique characterisability of linear temporal logic. Msc, University of Amsterdam, 2023.

[42] Muzammil Shahbaz and Roland Groz. Inferring mealy machines. In *International Symposium on Formal Methods*, pages 207–222. Springer, 2009.

[43] Slawek Staworko and Piotr Wieczorek. Characterizing XML twig queries with examples. In *Proc. of ICDT*, pages 144–160, 2015.

[44] Balder ten Cate and Victor Dalmau. Conjunctive queries: Unique characterizations and exact learnability. *ACM Trans. Database Syst.*, 47(4):14:1–14:41, 2022.

[45] Balder ten Cate, Víctor Dalmau, and Phokion G. Kolaitis. Learning schema mappings. *ACM Trans. Database Syst.*, 38(4):28:1–28:31, 2013.

[46] Balder ten Cate and Raoul Koudijs. Characterising modal formulas with examples. *CoRR*, abs/2304.06646, 2023.

[47] Stephan Tobies. *Complexity results and practical algorithms for logics in knowledge representation*. PhD thesis, RWTH Aachen University, Germany, 2001.

[48] Przemysław A. Wałęga, Bernardo Cuenca Grau, Mark Kaminski, and Egor V. Kostylev. DatalogMTL over the Integer Timeline. In *Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning*, pages 768–777, 9 2020.