Proceedings of the 19th International Conference on Principles of Knowledge Representation and Reasoning
Main Track

# Unique Characterisability and Learnability of Temporal Instance Queries

**Marie Fortin**[1] , **Boris Konev**[1] , **Vladislav Ryzhikov**[2] ,
**Yury Savateev**[2] , **Frank Wolter**[1] , **Michael Zakharyaschev**[2]
[1]Department of Computer Science, University of Liverpool, UK
[2]Department of Computer Science, Birkbeck, University of London, UK
{marie.fortin,boris.konev,wolter@liverpool.ac.uk}, {vlad,yury,michael}@dcs.bbk.ac.uk

**Abstract**

We aim to determine which temporal instance queries can be uniquely characterised by a (polynomial-size) set of positive and negative temporal data examples. We start by considering queries formulated in fragments of propositional linear temporal logic *LTL* that correspond to conjunctive queries (CQs) or extensions thereof induced by the until operator. Not all of these queries admit polynomial characterisations but by restricting them further to path-shaped queries we identify natural classes that do. We then investigate how far the obtained characterisations can be lifted to temporal knowledge graphs queried by 2D languages combining *LTL* with concepts in description logics $\mathcal{EL}$ or $\mathcal{ELI}$ (i.e., tree-shaped CQs). While temporal operators in the scope of description logic constructors can destroy polynomial characterisability, we obtain general transfer results for the case when description logic constructors are within the scope of temporal operators. Finally, we apply our characterisations to establish (polynomial) learnability of temporal instance queries using membership queries in the active learning framework.

## 1 Introduction

Constructing queries or, more generally, logical concepts describing individuals of interest, can be difficult. Providing support to a user to cope with this problem has been a major research topic in databases, logic, and knowledge representation. For instance, in reverse engineering of database queries and concept descriptions (Martins 2019; Lehmann and Hitzler 2010; Jung et al. 2020), one aims to identify a query using a set of positively and negatively labelled examples of answers and non-answers, respectively; and in active learning approaches, one aims to identify a query by asking an oracle (e.g., domain specialist) whether an example is an answer or a non-answer to the query (Angluin, Frazier, and Pitt 1992; Funk, Jung, and Lutz 2021; ten Cate and Dalmau 2021).

Recently, the *unique characterisation* of a query by a finite (ideally, polynomial-size) set of positive and negative example answers has been identified as a fundamental link between queries and data (ten Cate and Dalmau 2021). Namely, we say that a query $\boldsymbol{q}$ *fits* a pair $E = (E^+, E^-)$ of sets $E^+$ and $E^-$ of pointed databases $(\mathcal{D}, a)$ if $\mathcal{D} \models \boldsymbol{q}(a)$ for $(\mathcal{D}, a) \in E^+$, and $\mathcal{D} \not\models \boldsymbol{q}(a)$ for $(\mathcal{D}, a) \in E^-$. Then $E$ *uniquely characterises* $\boldsymbol{q}$ within a class $\mathcal{Q}$ of queries if $\boldsymbol{q}$ is the only (up to equivalence) query in $\mathcal{Q}$ that fits $E$.

Unique (polynomial) characterisations can be used to illustrate, explain, and construct queries. They are also a 'non-procedural' necessary condition for (polynomial) learnability using membership queries in Angluin's (1987b) framework of active learning, where membership queries to the oracle take the form 'does $\mathcal{D} \models \boldsymbol{q}(a)$ hold?'. It is shown by ten Cate and Dalmau (2021) that, for classes of conjunctive queries (CQs), it is often a sufficient condition as well.

In many applications, queries are required to capture the temporal evolution of individuals, making their formulation even harder. The aim of this paper is to start an investigation of the (polynomial) characterisability of temporal instance queries. We first consider one-dimensional data instances of the form $(\delta_0, \dots, \delta_n)$, where $\delta_i$ is the set of atomic propositions that are true at timestamp $i$, describing the temporal behaviour of a single individual, and queries formulated in fragments of propositional linear temporal logic *LTL*. Although rather basic as a temporal data model, this restriction allows us to focus on the purely temporal aspect of unique characterisability. We then generalise our results, where possible, to standard two-dimensional temporal data instances, in which the $\delta_i$ are replaced by non-temporal data instances and queries are obtained by combining fragments of *LTL* with $\mathcal{ELI}$-concept queries (or tree-shaped CQs), thereby combining a well established formalism for accessing temporal data (Chomicki and Toman 2018) with the basic concept descriptions for tractable data access from description logic (Baader et al. 2017).

Our initial observation is that already very primitive temporal queries are not uniquely characterisable. For example[1], consider the query $\boldsymbol{q} = \Diamond_r(A \wedge B)$ with the operator $\Diamond_r$ 'now or later' (interpreted by $\leq$ over linearly ordered timestamps). By the pigeonhole principle, no finite example set $E$ can distinguish $\boldsymbol{q}$ from a query $\boldsymbol{q}' = \Diamond_r(A \wedge (\Diamond_r B \wedge \Diamond_r(A \wedge \dots)))$ with sufficiently many alternating $A$ and $B$. Similarly, the query $\boldsymbol{q} = \bigcirc A$ with the 'next time' operator $\bigcirc$ is not distinguishable by a finite example set from $\boldsymbol{q}' = (\bigcirc \dots \bigcirc A) \,\mathsf{U}\, A$ with the strict 'until' operator $\mathsf{U}$ and sufficiently many $\bigcirc$ on its left-hand side.

Aiming to identify natural and useful classes of temporal queries enjoying (polynomial) characterisability, in this

[1]For detailed explanations and omitted proofs, the reader is referred to the full Arxiv version (Fortin et al. 2022b) of this paper.

paper we consider the conjunctive fragment of *LTL*. To begin with, we focus on two classes of path CQs: the class $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ of queries of the form

$$\boldsymbol{q} = \rho_0 \wedge \boldsymbol{o}_1(\rho_1 \wedge \boldsymbol{o}_2(\rho_2 \wedge \dots \wedge \boldsymbol{o}_n \rho_n)), \qquad (1)$$

where $\boldsymbol{o}_i \in \{\bigcirc, \Diamond_r\}$ and $\rho_i$ is a conjunction of atomic propositions, and also the class $\mathcal{Q}^\sigma_p[\mathsf{U}]$ of $\mathsf{U}$-queries of the form

$$\boldsymbol{q} = \rho_0 \wedge (\lambda_1 \mathsf{U} (\rho_1 \wedge (\lambda_2 \mathsf{U} (\dots (\lambda_n \mathsf{U} \rho_n) \dots)))), \qquad (2)$$

where $\lambda_i$ is a conjunction of atoms or $\bot$. The superscript $\sigma$ in $\mathcal{Q}^\sigma_p[\mathsf{U}]$ indicates that queries are formulated in a finite signature $\sigma$, a condition required because of the universal quantification implicit in $\mathsf{U}$. Our first main result is a syntactic criterion of (polynomial) characterisability of $\mathcal{Q}_p[\bigcirc, \Diamond_r]$-queries. In fact, it turns out that the query $\Diamond_r(A \wedge B)$ mentioned above epitomises the cause of non-characterisability in $\mathcal{Q}_p[\bigcirc, \Diamond_r]$. It follows, in particular, that the restriction $\mathcal{Q}_p[\bigcirc, \Diamond]$ of $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ to queries with $\bigcirc$ and strict eventuality $\Diamond = \bigcirc\Diamond_r$ is polynomially characterisable. Our second main result is that all $\mathcal{Q}^\sigma_p[\mathsf{U}]$-queries with $\subseteq$-incomparable $\lambda_i$ and $\rho_i$, for each $i$, are polynomially characterisable within $\mathcal{Q}^\sigma_p[\mathsf{U}]$. Although we show that all $\mathcal{Q}^\sigma_p[\mathsf{U}]$-queries are exponentially characterisable, it remains open whether they are polynomially characterisable in $\mathcal{Q}^\sigma_p[\mathsf{U}]$.

The essential property that distinguishes $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ and $\mathcal{Q}^\sigma_p[\mathsf{U}]$ from other queries is that they do not admit temporal branching as, for instance, in $\Diamond A \wedge \Diamond B$. In fact, we show that even within the class of queries using only $\wedge$ and $\Diamond$ and with a bound on the number of branches, not all queries are polynomially characterisable. A first step towards positive results covering non-path queries is made for the case of queries in which all branches are of equal length.

Our next aim is to generalise the obtained results to 2D temporal queries combining *LTL* with the description logic constructor $\exists P$ of $\mathcal{ELI}$. Our first main result is negative: even queries of the form $\exists P.\boldsymbol{q}_1 \wedge \dots \wedge \exists P.\boldsymbol{q}_n$, in which $\boldsymbol{q}_i \in \mathcal{Q}_p[\bigcirc, \Diamond]$, are not polynomially characterisable. The situation changes drastically, however, if we consider queries of the form (1) or (2), in which $\rho_i$ and $\lambda_i$ are $\mathcal{ELI}$-queries. Indeed, we generalise our polynomial characterisability results for $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ and $\mathcal{Q}^\sigma_p[\mathsf{U}]$ to such queries using recent results on the computation of frontiers in the lattice of $\mathcal{ELI}$-queries (ten Cate and Dalmau 2021) and proving a new result on split partners in the lattice of $\mathcal{EL}$-queries (where $\mathcal{EL}$ denotes $\mathcal{ELI}$ without inverse roles).

Finally, we discuss applications of our results to learning temporal instance queries using membership queries of the form 'does $\mathcal{D} \models \boldsymbol{q}$ hold?'. 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. A main result here is that $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ with $\mathcal{ELI}$-queries is polynomial-time learnable with membership queries, assuming the learner is given the target query size in advance.

## 2 Related Work

Our contribution is closely related to work on active learnability of formal languages and on learning temporal logic formulas interpreted over finite and infinite traces. It is also related to learning database queries and other formal expressions for accessing data. In the former area, the seminal paper by Angluin (1987a) has given rise to a large body of work on active learning of regular languages or variations, for example, (Shahbaz and Groz 2009; Aarts and Vaandrager 2010; Cassel et al. 2016; Howar and Steffen 2018). This work has mainly focused on learning various types of finite state machines or automata 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 a wide variety of important languages, including regular languages. In fact, the main difference between this work and our contribution is that we focus on queries for which the corresponding formal languages form only a small subset of the regular languages and it is this restriction that enables us to focus on characterisability and learnability with membership queries.

Rather surprisingly, there has hardly been any work on active learning of temporal formulas over finite or infinite traces; we refer the reader to (Camacho and McIlraith 2019), also for a discussion of the relationship between learning automata and *LTL*-formulas. In contrast, passive learning of *LTL*-formulas has recently received significant attention; see (Lemieux, Park, and Beschastnikh 2015; Neider and Gavran 2018; Fijalkow and Lagarde 2021; Fortin et al. 2022a) and, in the context of explainable AI, also (Camacho and McIlraith 2019) for an overview.

In the database and KR communities, there has been extensive work on identifying queries and concept descriptions from data examples. 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 (Arenas and Diaz 2016; Barceló and Romero 2017) and under the open world assumption (Gutiérrez-Basulto, Jung, and Sabellek 2018; Funk et al. 2019). Related work on active learning not yet discussed include the identification of $\mathcal{EL}$-queries (Funk, Jung, and Lutz 2021) and ontologies (Konev, Ozaki, and Wolter 2016; Konev et al. 2017), and of schema-mappings (ten Cate, Dalmau, and Kolaitis 2013; ten Cate et al. 2018). Again this work differs from our contribution as it focuses on learning using membership and equivalence queries rather than only the former. The use of unique characterisations to explain and construct schema mappings has been promoted and investigated by Kolaitis (2011) and Alexe et al. (2011).

Combining *LTL* and description logics for temporal conceptual modelling and data access has a long tradition (Lutz, Wolter, and Zakharyaschev 2008; Artale et al. 2017). For querying purposes, sometimes description logic concepts have been replaced by general CQs. Our restriction to $\mathcal{ELI}$-concepts instead of general CQs is motivated by results of (ten Cate and Dalmau 2021) showing that only CQs that are acyclic modulo cycles through the answer variables are polynomially characterisable within the class of CQs. Hence very strong acyclicity conditions are needed to ensure polynomial characterisability. We conjecture that our results can be extended to this class.

The class of queries in which no $\exists P$ is within the scope of temporal operators was first introduced by (Baader, Borgwardt, and Lippmann 2015; Borgwardt and Thost 2015) in the context of monitoring applications. The lcs and msc in temporal DLs are considered by Tirtarasa and Turhan 2022.

## 3 Preliminaries

By a *signature* we mean any finite set $\sigma \neq \emptyset$ of *atomic concepts* $A, B, C, \dots$ representing observations, measurements, events, etc. A *$\sigma$-data instance* is any finite sequence $\mathcal{D} = (\delta_0, \dots, \delta_n)$ with $\delta_i \subseteq \sigma$, saying that $A \in \delta_i$ happened at moment $i$. The *length* of $\mathcal{D}$ is $\max(\mathcal{D}) = n$ and the *size* of $\mathcal{D}$ is $|\mathcal{D}| = \Sigma_{i \le n} |\delta_i|$. We do not distinguish between $\mathcal{D}$ and its variants of the form $(\delta_0, \dots, \delta_n, \emptyset, \dots, \emptyset)$.

We access data by means of *queries*, $\boldsymbol{q}$, constructed from atoms, $\bot$ and $\top$ using $\wedge$ and the temporal operators $\bigcirc$, $\Diamond$, $\Diamond_r$ and $\mathsf{U}$. The set of atomic concepts occurring in $\boldsymbol{q}$ is denoted by $sig(\boldsymbol{q})$. The set of queries that only use the operators from $\Phi \subseteq \{\bigcirc, \Diamond, \Diamond_r, \mathsf{U}\}$ is denoted by $\mathcal{Q}[\Phi]$; $\mathcal{Q}^\sigma[\Phi]$ is the restriction of $\mathcal{Q}[\Phi]$ to a signature $\sigma$. 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 $\boldsymbol{q}$.

$\mathcal{Q}[\bigcirc, \Diamond, \Diamond_r]$-queries can be equivalently defined as *tree-shaped* conjunctive queries (CQs) with the binary predicates *suc*, $<$, $\le$ over $\mathbb{N}$, and atomic concepts as unary predicates. Each such CQ is a set $Q(t_0)$ of assertions of the form $A(t)$, $suc(t, t')$, $t < t'$, and $t \le t'$, with a distinguished variable $t_0$, such that, for every variable $t$ in $Q(t_0)$, there exists exactly one path from $t_0$ to $t$ along the binary predicates *suc*, $<$, $\le$.

The set of $\mathcal{Q}[\bigcirc, \Diamond, \Diamond_r]$-queries with *path-shaped* CQ counterparts is denoted by $\mathcal{Q}_p[\bigcirc, \Diamond, \Diamond_r]$. Such queries $\boldsymbol{q}$ take the form (1), where $\boldsymbol{o}_i \in \{\bigcirc, \Diamond, \Diamond_r\}$ and $\rho_i$ is a conjunction of atoms (the empty conjunction is $\top$). Similarly, $\mathcal{Q}_p[\mathsf{U}]$-queries take the form (2).

Given a data instance $\mathcal{D} = (\delta_0, \dots, \delta_n)$, the *truth-relation* $\mathcal{D}, \ell \models \boldsymbol{q}$, for $\ell < \omega$, is defined as follows:

$$\mathcal{D}, \ell \models A \text{ iff } A \in \delta_\ell, \qquad \mathcal{D}, \ell \models \top, \qquad \mathcal{D}, \ell \not\models \bot,$$
$$\mathcal{D}, \ell \models \boldsymbol{q}_1 \wedge \boldsymbol{q}_2 \text{ iff } \mathcal{D}, \ell \models \boldsymbol{q}_1 \text{ and } \mathcal{D}, \ell \models \boldsymbol{q}_2,$$
$$\mathcal{D}, \ell \models \bigcirc \boldsymbol{q} \text{ iff } \mathcal{D}, \ell + 1 \models \boldsymbol{q},$$
$$\mathcal{D}, \ell \models \Diamond \boldsymbol{q} \text{ iff } \mathcal{D}, m \models \boldsymbol{q}, \text{ for some } m > \ell,$$
$$\mathcal{D}, \ell \models \Diamond_r \boldsymbol{q} \text{ iff } \mathcal{D}, m \models \boldsymbol{q}, \text{ for some } m \ge \ell,$$
$$\mathcal{D}, \ell \models \boldsymbol{q}_1 \mathsf{U} \boldsymbol{q}_2 \text{ iff there is } m > \ell \text{ such that } \mathcal{D}, m \models \boldsymbol{q}_2 \text{ and } \mathcal{D}, k \models \boldsymbol{q}_1, \text{ for all } k \text{ with } \ell < k < m.$$

Note that $\mathcal{D}, n \models \Diamond\top \wedge \bigcirc\top \wedge (\boldsymbol{q} \mathsf{U} \top)$ as $(\delta_0, \dots, \delta_n, \emptyset)$ is a variant of $\mathcal{D}$. We write $\boldsymbol{q} \models \boldsymbol{q}'$ if $\mathcal{D}, \ell \models \boldsymbol{q}$ implies $\mathcal{D}, \ell \models \boldsymbol{q}'$ for any $\mathcal{D}$ and $\ell$. If $\boldsymbol{q} \models \boldsymbol{q}'$ and $\boldsymbol{q}' \models \boldsymbol{q}$, we call $\boldsymbol{q}$ and $\boldsymbol{q}'$ *equivalent* and write $\boldsymbol{q} \equiv \boldsymbol{q}'$. Since $\bigcirc\boldsymbol{q} \equiv \bot \mathsf{U} \boldsymbol{q}$, $\Diamond\boldsymbol{q} \equiv \top \mathsf{U} \boldsymbol{q}$ and $\Diamond\boldsymbol{q} \equiv \bigcirc\Diamond_r\boldsymbol{q}$, one can assume that $\mathcal{Q}[\bigcirc, \Diamond] \subseteq \mathcal{Q}[\mathsf{U}]$, $\mathcal{Q}[\Diamond] \subseteq \mathcal{Q}[\bigcirc, \Diamond_r]$ and $\mathcal{Q}[\bigcirc, \Diamond_r] = \mathcal{Q}[\bigcirc, \Diamond_r, \Diamond]$.

## 4 Unique Characterisability

An *example set* is a pair $E = (E^+, E^-)$ with finite sets $E^+$ and $E^-$ of data instances that are called *positive* and *negative examples*, respectively. A query $\boldsymbol{q}$ *fits* $E$ if $\mathcal{D}^+, 0 \models \boldsymbol{q}$ and $\mathcal{D}^-, 0 \not\models \boldsymbol{q}$, for all $\mathcal{D}^+ \in E^+$ and $\mathcal{D}^- \in E^-$. We say that $E$ *uniquely characterises* $\boldsymbol{q}$ within a class $\mathcal{Q}$ of queries if $\boldsymbol{q}$ fits $E$ and $\boldsymbol{q} \equiv \boldsymbol{q}'$ for any $\boldsymbol{q}' \in \mathcal{Q}$ that fits $E$. If all $\boldsymbol{q} \in \mathcal{Q}$ are characterised by some $E$ within $\mathcal{Q}' \supseteq \mathcal{Q}$, we call $\mathcal{Q}$ *uniquely characterisable* within $\mathcal{Q}'$. Further, $\mathcal{Q}$ is *polynomially characterisable* within $\mathcal{Q}' \supseteq \mathcal{Q}$ if there is a polynomial $f$ such that every $\boldsymbol{q} \in \mathcal{Q}$ is characterised within $\mathcal{Q}'$ by some $E$ of size $|E| \le f(|\boldsymbol{q}|)$, where $|E| = \Sigma_{\mathcal{D} \in (E^+ \cup E^-)} |\mathcal{D}|$. Let $\mathcal{Q}^n$ be the set of queries in $\mathcal{Q}$ of size at most $n$. We say that $\mathcal{Q}$ is *polynomially characterisable for bounded query size* if there is a polynomial $f$ such that every $\boldsymbol{q} \in \mathcal{Q}^n$ is characterised by some $E$ of size $\le f(n)$ within $\mathcal{Q}^n$.

Observe that (polynomial) characterisability is anti-monotone: if a query $\boldsymbol{q}$ is (polynomially) characterisable within $\mathcal{Q}$ and $\mathcal{Q}' \subseteq \mathcal{Q}$, then $\boldsymbol{q}$ is (polynomially) characterisable within $\mathcal{Q}'$. In counterexamples to characterisability, we therefore only provide the smallest natural class of queries within which non-characterisability holds. The following examples illustrate (non-)characterisability within the classes $\mathcal{Q}_p[\Diamond_r]$ and $\mathcal{Q}_p[\bigcirc, \Diamond_r]$.

**Example 1.** $(i)$ Recall from Section 1 that $\Diamond_r(A \wedge B)$ is not uniquely characterisable within $\mathcal{Q}_p[\Diamond_r]$. The same argument shows non-characterisability of $\Diamond(A \wedge B)$ within $\mathcal{Q}_p[\Diamond_r, \Diamond]$. On the other hand, the query $\Diamond(A \wedge B)$ is characterised within $\mathcal{Q}_p[\Diamond, \bigcirc]$ by the example set with positive examples $(\emptyset, \{A, B\})$ and $(\emptyset, \emptyset, \{A, B\})$ and negative examples $(\emptyset, \{A\})$ and $(\emptyset, \{B\})$.

$(ii)$ The conjunction of atoms does not always lead to non-characterisability within classes of queries with $\Diamond_r$. For example, $\boldsymbol{q} = \Diamond_r(A \wedge \bigcirc(A \wedge B))$ is characterised within $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ by $E = (E^+, E^-)$ in which $E^+$ contains two data instances $(\{A\}, \{A, B\})$ and $(\emptyset, \{A\}, \{A, B\})$ and $E^-$ also two instances:

$$(\emptyset, \emptyset, \{A, B\}), \quad (\emptyset, \{A\}, \{A\}, \{B\}, \{A, B\}).$$

The intuition here is that some instances from $E^-$ have to satisfy the query $\Diamond_r(A \wedge \bigcirc(B \wedge \Diamond_r(A \wedge B)))$ as well as the query $\Diamond_r(A \wedge \bigcirc(A \wedge \Diamond_r(A \wedge B)))$.

$(iii)$ While the query $\Diamond_r(A \wedge B)$ from $(i)$ is not characterisable, there is a polynomial $f$ such that, for all $n \in \mathbb{N}$, it is characterisable within $\mathcal{Q}_p^n[\bigcirc, \Diamond_r]$ by some $E_n$ of size $\le f(n)$. Namely, we take $E^+ = \{(\{A, B\}), (\emptyset, \{A, B\})\}$ and $E^- = \{(\underbrace{\{A\}, \{B\}, \dots, \{A\}, \{B\}}_{n \text{ times}})\}$.

Observe that one can always separate $\boldsymbol{q} \in \mathcal{Q}[\bigcirc, \Diamond_r]$ from any other $\boldsymbol{q}' \in \mathcal{Q}[\bigcirc, \Diamond_r]$ with $sig(\boldsymbol{q}') \supsetneq \mathrm{sig}(\boldsymbol{q}) = \sigma$ using the positive example $(\sigma, \dots, \sigma)$ with $tdp(\boldsymbol{q})+1$-many copies of $\sigma$. One can therefore focus on characterisability within the relevant class of queries over the same signature as the input query. However, this is not the case for $\mathcal{Q}[\mathsf{U}]$:

**Example 2.** The query $\boldsymbol{q} = \bot \mathsf{U} A \equiv \bigcirc A$ is not uniquely characterisable within $\mathcal{Q}_p[\mathsf{U}]$. Indeed, suppose $\boldsymbol{q}$ fits $E$ and $\sigma$ comprises all atoms occurring in $E$. Then $\mathcal{D}, 0 \models C \mathsf{U} A$ iff $\mathcal{D}, 0 \models \bigcirc A$, for all $\mathcal{D}$ in $E$ and $C \notin \sigma$, and so $E$ does not characterise $\boldsymbol{q}$. On the other hand, for the signature $\sigma = \{A, B\}$, the query $\boldsymbol{q}$ is characterised within $\mathcal{Q}_p^\sigma[\mathsf{U}]$ by the example set $(E^+, E^-)$ in which $E^+ = \{(\emptyset, \{A\})\}$ and $E^- = \{(\sigma, \{B\}, \{A\})\}$ as $A \mathsf{U} A \equiv (A \wedge B) \mathsf{U} A \equiv \bigcirc A$.

As noted in Section 1, $\bot \mathsf{U} A$ is not uniquely characterisable within $\mathcal{Q}^{\{A\}}[\mathsf{U}]$ because of nested U-operators on the left-hand side of U. This observation prompts us to consider the subclass $\mathcal{Q}^{\sigma}_{-}[\mathsf{U}]$ of $\mathcal{Q}^{\sigma}[\mathsf{U}]$-queries $\boldsymbol{q}$ in which any subquery $\boldsymbol{q}' \mathsf{U} \boldsymbol{q}''$ does not contain occurrences of U in $\boldsymbol{q}'$. Note that $\mathcal{Q}^{\sigma}_{p}[\mathsf{U}] \subseteq \mathcal{Q}^{\sigma}_{-}[\mathsf{U}]$. We show that $\mathcal{Q}^{\sigma}_{-}[\mathsf{U}]$ is uniquely characterisable. To simplify notation, we give $\sigma$-data instances as *words* over the alphabet $2^{\sigma}$ using the standard notation of regular languages. Instead of $\mathcal{D}, 0 \models \boldsymbol{q}$ we simply write $\mathcal{D} \models \boldsymbol{q}$. By the semantics of U, for any $\boldsymbol{q} \in \mathcal{Q}^{\sigma}_{-}[\mathsf{U}]$, we have

$$\sigma^d \not\models \boldsymbol{q} \ \text{ for } d \leq \mathit{tdp}(\boldsymbol{q}), \quad \sigma^d \models \boldsymbol{q} \ \text{ for } d > \mathit{tdp}(\boldsymbol{q}) \tag{3}$$

where $\sigma^d$ is a word with $d$-many $\sigma$. Note also that there are finitely-many, say $N_d < \omega$, pairwise non-equivalent queries of any depth $d < \omega$ in $\mathcal{Q}^{\sigma}_{-}(\mathsf{U})$.

**Lemma 3.** *If $\boldsymbol{q}, \boldsymbol{q}' \in \mathcal{Q}^{\sigma}_{-}[\mathsf{U}]$ are of depth d and $\boldsymbol{q} \not\models \boldsymbol{q}'$, then there is $\mathcal{D}$ such that $\max(\mathcal{D}) \leq N_d$, $\mathcal{D} \models \boldsymbol{q}$ and $\mathcal{D} \not\models \boldsymbol{q}'$.*

*Proof.* Consider $\mathcal{D}$ of minimal length such that $\mathcal{D} \models \boldsymbol{q}$ and $\mathcal{D} \not\models \boldsymbol{q}'$. Let $\mathit{tp}(i)$ comprise all of the subqueries $\boldsymbol{s}$ of $\boldsymbol{q}$ and $\boldsymbol{q}'$ with $\mathcal{D}, i \models \boldsymbol{s}$. By the choice of $\mathcal{D}$, we have $\mathit{tp}(i) \neq \mathit{tp}(j)$ for any distinct $i, j \in [0, \max(\mathcal{D})]$ (otherwise we could cut the interval $[i, j)$ out of $\mathcal{D}$ to obtain a shorter instance separating $\boldsymbol{q}$ from $\boldsymbol{q}'$). It follows that $\max(\mathcal{D}) \leq N_d$. ❑

**Theorem 4.** *For any $\sigma$, $\mathcal{Q}^{\sigma}_{-}[\mathsf{U}]$ is uniquely characterisable.*

*Proof.* Any $\boldsymbol{q} \in \mathcal{Q}^{\sigma}_{-}(\mathsf{U})$ is uniquely characterised by $E$ with

$$E^+ = \{\mathcal{D} \models \boldsymbol{q} \mid \max(\mathcal{D}) \leq N_{\mathit{tdp}(\boldsymbol{q})}\},$$
$$E^- = \{\mathcal{D} \not\models \boldsymbol{q} \mid \max(\mathcal{D}) \leq N_{\mathit{tdp}(\boldsymbol{q})}\}.$$

Indeed, let $\boldsymbol{q}' \in \mathcal{Q}^{\sigma}_{-}(\mathsf{U})$ fit $E$. Then $\mathit{tdp}(\boldsymbol{q}') = \mathit{tdp}(\boldsymbol{q})$ by (3), and so $\boldsymbol{q} \equiv \boldsymbol{q}'$ by Lemma 3. ❑

It follows from the proof that $\mathcal{Q}[\bigcirc, \Diamond]$ is uniquely characterisable as well.

## 5 Characterisability in $\mathcal{Q}_p[\bigcirc, \Diamond_r]$

In this section, we prove a criterion of (polynomial) unique characterisability of queries within $\mathcal{Q}_p[\bigcirc, \Diamond_r]$. The criterion is applicable to $\mathcal{Q}_p[\bigcirc, \Diamond, \Diamond_r]$-queries in a normal form, which is defined and illustrated below.

**Example 5.** It is readily checked that the $\mathcal{Q}_p[\bigcirc, \Diamond_r]$-query $\boldsymbol{q} = \bigcirc\Diamond_r\bigcirc\Diamond_r(A \land B \land C \land \Diamond_r(B \land \Diamond_r(B \land C)))$ is equivalent to the $\mathcal{Q}_p[\bigcirc, \Diamond, \Diamond_r]$-query $\boldsymbol{q}^{\mathit{nf}} = \Diamond\Diamond(A \land B \land C)$.

We define the normal form for $\mathcal{Q}_p[\bigcirc, \Diamond, \Diamond_r]$-queries represented as a first-order CQ by a list of atoms. For example, the query $\boldsymbol{q}^{\mathit{nf}}$ above is given by the CQ

$$\boldsymbol{q}^{\mathit{nf}}(t_0) = t_0 < t_1, t_1 < t_2, A(t_2), B(t_2), C(t_2)$$

with one free (answer) variable $t_0$ and existentially quantified $t_1$ and $t_2$. In general, any $\boldsymbol{q} \in \mathcal{Q}_p[\bigcirc, \Diamond, \Diamond_r]$ is represented as a CQ

$$\rho_0(t_0), R_1(t_0, t_1), \dots, \rho_{n-1}(t_{n-1}), R_n(t_{n-1}, t_n), \rho_n(t_n),$$

where $\rho_i$ is a set of atoms, $\rho_i(t_i) = \{A(t_i) \mid A \in \rho_i\}$ and $R_i \in \{\mathit{suc}, <, \leq\}$. We divide $\boldsymbol{q}$ into *blocks* $\boldsymbol{q}_i$ such that

$$\boldsymbol{q} = \boldsymbol{q}_0 \mathcal{R}_1 \boldsymbol{q}_1 \dots \mathcal{R}_n \boldsymbol{q}_n \tag{4}$$

with $\mathcal{R}_i = R^i_1(t^i_0, t^i_1) \dots R^i_{n_i}(t^i_{n_i-1}, t^i_{n_i})$, for $R^i_j \in \{<, \leq\}$,

$$\boldsymbol{q}_i = \rho^i_0(s^i_0)\mathit{suc}(s^i_0, s^i_1)\rho^i_1(s^i_1) \dots \mathit{suc}(s^i_{k_i-1}, s^i_{k_i})\rho^i_{k_i}(s^i_{k_i})$$

and $s^i_{k_i} = t^{i+1}_0$, $t^i_{n_i} = s^i_0$. If $k_i = 0$, the block $\boldsymbol{q}_i$ is *primitive*. A primitive block $\boldsymbol{q}_i = \rho^i_0(s^i_0)$ with $i > 0$ and $|\rho^i_0| \geq 2$ is called a *lone conjunct* of $\boldsymbol{q}$.

**Example 6.** The query $\Diamond_r(A \land B)$ in Example 1$(i)$, whose CQ representation is $t_0 \leq t_1, \rho_1(t_1)$, for $\rho_1 = \{A, B\}$, has a lone conjunct $\rho_1(t_1)$. In $\Diamond_r(A \land \bigcirc(A \land B))$ from Example 1$(ii)$, represented as $t_0 \leq t_1$, $A(t_1), \mathit{suc}(t_1, t_2), \rho_1(t_2)$, the conjunct $\rho_1(t_2)$ is not lone.

Now, we say that $\boldsymbol{q}$ given by (4) is in *normal form* if the following conditions are satisfied:

**(n1)** $\rho^i_0 \neq \emptyset$ if $i > 0$, and $\rho^i_{k_i} \neq \emptyset$ if $i > 0$ or $k_i > 0$ (thus, of all the first/last $\rho$ in a block, only $\rho^0_0$ can be empty);

**(n2)** each $\mathcal{R}_i$ is either a single $t^i_0 \leq t^i_1$ or a sequence of $<$;

**(n3)** $\rho^i_{k_i} \not\supseteq \rho^{i+1}_0$ if $\boldsymbol{q}_{i+1}$ is primitive and $R_{i+1}$ is $\leq$;

**(n4)** $\rho^i_{k_i} \not\subseteq \rho^{i+1}_0$ if $i > 0$, $\boldsymbol{q}_i$ is primitive and $R_{i+1}$ is $\leq$.

The queries in Example 6 are in normal form with two blocks each; the query $\boldsymbol{q}^{\mathit{nf}}$ above is in normal form with two blocks $\boldsymbol{q}_0 = \top(t_0)$ and $\boldsymbol{q}_1 = A(t_2) \land B(t_2) \land C(t_2)$.

**Lemma 7.** *Every query in $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ is equivalent to a query in normal form that can be computed in linear time.*

A query $\boldsymbol{q} \in \mathcal{Q}_p[\bigcirc, \Diamond]$ is *safe* if it is equivalent to a query $\boldsymbol{q}' \in \mathcal{Q}_p[\bigcirc, \Diamond]$ in normal form not containing lone conjuncts. We are now in the position to formulate the criterion.

**Theorem 8.** $(i)$ *A query $\boldsymbol{q} \in \mathcal{Q}_p[\bigcirc, \Diamond_r]$ is uniquely characterisable within $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ iff $\boldsymbol{q}$ is safe.*

$(ii)$ *Those queries that are uniquely characterisable within $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ are actually polynomially characterisable within $\mathcal{Q}_p[\bigcirc, \Diamond_r]$.*

$(iii)$ *The class $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ is polynomially characterisable for bounded query size.*

$(iv)$ *The class $\mathcal{Q}_p[\bigcirc, \Diamond]$ is polynomially characterisable.*

*Proof sketch.* A detailed proof is given in the full version. Here, we define a polysize example set $E = (E^+, E^-)$ characterising a query $\boldsymbol{q}$ in normal form (4), which does not contain lone conjuncts. Let $b$ be the number of $\bigcirc$ and $\Diamond$ in $\boldsymbol{q}$ plus 1. For each block $\boldsymbol{q}_i$ in (4), we take two words

$$\bar{\boldsymbol{q}}_i = \rho^i_0 \dots \rho^i_{k_i}, \quad \bar{\boldsymbol{q}}_i \bowtie \bar{\boldsymbol{q}}_{i+1} = \rho^i_0 \dots (\rho^i_{k_i} \cup \rho^{i+1}_0) \dots \rho^{i+1}_{k_{i+1}}.$$

The set $E^+$ contains the data instances given by the words

- $\mathcal{D}_b = \bar{\boldsymbol{q}}_0 \emptyset^b \dots \bar{\boldsymbol{q}}_i \emptyset^b \bar{\boldsymbol{q}}_{i+1} \dots \emptyset^b \bar{\boldsymbol{q}}_n$,
- $\mathcal{D}_i = \bar{\boldsymbol{q}}_0 \emptyset^b \dots \bar{\boldsymbol{q}}_i \bowtie \bar{\boldsymbol{q}}_{i+1} \dots \emptyset^b \bar{\boldsymbol{q}}_n$ 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 also the data instances obtained from $\mathcal{D}_b$ by

(a) removing a single atom from some $\rho^i_j \neq \emptyset$ or removing the whole $\rho^i_j = \emptyset$, for $i \neq 0$ and $j \neq 0$, from some $\bar{\boldsymbol{q}}_i$;

(b) replacing $\bar{\boldsymbol{q}}_i = \rho^i_0 \dots \rho^i_l \rho^i_{l+1} \dots \rho^i_{k_i}$ ($k_i > 0$) by $\bar{\boldsymbol{q}}'_i \emptyset^b \bar{\boldsymbol{q}}''_i$, where $\bar{\boldsymbol{q}}'_i = \rho^i_0 \dots \rho^i_l$, $\bar{\boldsymbol{q}}''_i = \rho^i_{l+1} \dots \rho^i_{k_i}$ and $l \geq 0$;

(c) replacing some $\rho^i_l \neq \emptyset$, $0 < l < k_i$, by $\rho^i_l \emptyset^b \rho^i_l$;

(d) replacing $\rho^i_{k_i}$ ($k_i > 0$, $|\rho^i_{k_i}| \geq 2$) with $\rho^i_{k_i} \setminus \{A\} \emptyset^b \rho^i_{k_i}$, for some $A \in \rho^i_{k_i}$, or replacing $\rho^i_0$ ($k_i > 0$, $|\rho^i_0| \geq 2$) with $\rho^i_0 \emptyset^b \rho^i_0 \setminus \{A\}$, for some $A \in \rho^i_0$;

(e) replacing $\rho^0_0 \neq \emptyset$ with $\rho^0_0 \setminus \{A\} \emptyset^b \rho^0_0$, for some $A \in \rho^0_0$, if $k_0 = 0$, and with $\rho^0_0 \emptyset^b \rho^0_0$ if $k_0 > 0$.

The size of $E$ is clearly polynomial in $|\boldsymbol{q}|$. It is readily seen that $\mathcal{D} \models \boldsymbol{q}$ for all $\mathcal{D} \in E^+$. To continue the proof sketch, note that $\mathcal{D} \models \boldsymbol{q}$ iff there is a *homomorphism* $h$ from the set $\mathit{var}(\boldsymbol{q})$ of variables in $\boldsymbol{q}$ to $[0, \max(\mathcal{D})]$, i.e., $h(t_0) = 0$, $A(h(t)) \in \mathcal{D}$ if $A(t) \in \boldsymbol{q}$, $h(t') = h(t) + 1$ if $\mathit{suc}(t, t') \in \boldsymbol{q}$, and $h(t) \, R \, h(t')$ if $R(t, t') \in \boldsymbol{q}$ for $R \in \{<, \leq\}$. Using the assumption that $\boldsymbol{q}$ is in normal form, one can show that there is no homomorphism witnessing $\mathcal{D} \models \boldsymbol{q}$, for any $\mathcal{D} \in E^-$.

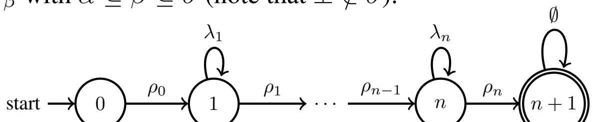
Suppose now that $\boldsymbol{q}' \in \mathcal{Q}_p[\bigcirc, \Diamond_r]$ in normal form is given and $\boldsymbol{q}' \not\equiv \boldsymbol{q}$. If $\mathcal{D}_b \not\models \boldsymbol{q}'$, we are done as $\mathcal{D}_b \in E^+$. Otherwise, let $h$ be a homomorphism witnessing $\mathcal{D}_b \models \boldsymbol{q}'$. Then one can show that either the restriction of $h$ to the blocks of $\boldsymbol{q}'$ is an isomorphism onto the blocks of $\boldsymbol{q}$ or there exists a data instance $\mathcal{D}$ obtained using one of the rules (a)–(e) such that a suitably modified $h$ is a homomorphism from $\boldsymbol{q}'$ to $\mathcal{D}$. In the latter case, we are done as $\mathcal{D} \in E^-$ and $\mathcal{D} \models \boldsymbol{q}'$. In the former case, $\boldsymbol{q}$ and $\boldsymbol{q}'$ coincide with the exception of the sequences of $\Diamond$ and $\Diamond_r$ between blocks. Then $\boldsymbol{q}$ can be separated from $\boldsymbol{q}'$ using the examples $\mathcal{D}_i$ and $\mathcal{D}^-_i$. ❑

## 6 Polynomial Characterisability in $\mathcal{Q}^\sigma_p[\mathsf{U}]$

*LTL*-queries with $\mathsf{U}$ do not correspond to CQs (because of the universal quantification in its semantics), and so require a different approach. We view them as defining regular languages. With each $\mathcal{Q}^\sigma_p[\mathsf{U}]$-query of the form (2) we associate the following regular expression over the alphabet $2^\sigma$:

$$\boldsymbol{q} = \rho_0 \lambda^*_1 \rho_1 \lambda^*_2 \dots \lambda^*_n \rho_n \lambda^*_{n+1} \tag{5}$$

where $\lambda_{n+1} = \emptyset$ and $\bot^* = \varepsilon$. We regard the words of the language $\boldsymbol{L}(\boldsymbol{q})$ over $2^\sigma$ as data instances. Clearly, $\mathcal{D}' \models \boldsymbol{q}$ iff there is $\mathcal{D} \in \boldsymbol{L}(\boldsymbol{q})$ such that $\mathcal{D} \Subset \mathcal{D}'$, i.e., $\mathcal{D} = (\delta_0, \dots, \delta_k)$ and $\mathcal{D}' = (\delta'_0, \dots, \delta'_k)$, for some $k < \omega$, and $\delta_i \subseteq \delta'_i$, for all $i \leq k$. The language $\boldsymbol{L_q}$ of all $\sigma$-data instances $\mathcal{D} \models \boldsymbol{q}$ (regarded as words over $2^\sigma$) can be given by the NFA $\mathfrak{A}_{\boldsymbol{q}}$ below, where each $\to_\alpha$, for $\alpha \neq \bot$, stands for all transitions $\to_\beta$ with $\alpha \subseteq \beta \subseteq \sigma$ (note that $\bot \notin \sigma$):




Without loss of generality we assume that all our $\boldsymbol{q}$ are *minimal* in the sense that by replacing any $\lambda_i \neq \bot$ with $\bot$ in $\boldsymbol{q}$ we obtain a query that is *not equivalent* to $\boldsymbol{q}$. For example, in minimal $\boldsymbol{q}$, $\rho_j \supseteq \dots \supseteq \rho_i \supseteq \lambda_i$ and $\lambda_l = \bot$ for all $l \in (j, i)$ imply $\rho_j \not\subseteq \lambda_j$ as otherwise $\lambda_j \, \mathsf{U} \, (\rho_j \wedge (\bot \, \mathsf{U} \dots (\lambda_i \, \mathsf{U} \, \varphi) \dots))$ is equivalent to $\bot \, \mathsf{U} \, (\rho_j \wedge (\bot \, \mathsf{U} \dots (\lambda_i \, \mathsf{U} \, \varphi) \dots))$. Using standard automata-theoretic techniques, one can show:

**Theorem 9.** *Any $\mathcal{Q}^\sigma_p[\mathsf{U}]$-queries $\boldsymbol{q} \not\equiv \boldsymbol{q}'$ can be separated by some $\mathcal{D}$ with $\max(\mathcal{D}) \leq O((\min\{\mathit{tdp}(\boldsymbol{q}), \mathit{tdp}(\boldsymbol{q}')\})^2)$.*

Using Theorem 9 in the proof of Theorem 4 we obtain:

**Corollary 10.** *The class $\mathcal{Q}^\sigma_p[\mathsf{U}]$ is exponentially characterisable within $\mathcal{Q}^\sigma_p[\mathsf{U}]$.*

The following examples illustrate difficulties in finding short unique characterisations of $\mathcal{Q}^\sigma_p[\mathsf{U}]$-queries, namely, that in general, data instances of different shapes and forms are needed to separate $\mathcal{Q}^\sigma_p[\mathsf{U}]$-queries. To unclutter notation we omit $\{\}$ in singletons like $\{A\}$.

**Example 11.** (a) The shortest data instance separating

$$\boldsymbol{q} = X\emptyset^* A \bot^* B \bot^* AB^* AA^* B \emptyset^*,$$
$$\boldsymbol{q}' = X\emptyset^* A \bot^* BA^* AB^* A \bot^* B \emptyset^*$$

is $\mathcal{D} = XABABBAAB$ with $\mathcal{D} \models \boldsymbol{q}$ and $\mathcal{D} \not\models \boldsymbol{q}'$ (e.g., $XABABAAB$ satisfies both $\boldsymbol{q}$ and $\boldsymbol{q}'$).

(b) For $l > 0$, let $\boldsymbol{q}_l = (AB^*)^{l-1} AA^* BB^*$. Then

$$XA^* \boldsymbol{q}_{l_1} \boldsymbol{q}_{l_2} \dots \boldsymbol{q}_{l_k} X \emptyset^* \not\equiv X \bot^* \boldsymbol{q}_{l_1} \boldsymbol{q}_{l_2} \dots \boldsymbol{q}_{l_k} X \emptyset^*,$$
$$XA^* \boldsymbol{q}_{l_1} \boldsymbol{q}_{l_2} \dots \boldsymbol{q}_{l_k} A \emptyset^* \equiv X \bot^* \boldsymbol{q}_{l_1} \boldsymbol{q}_{l_2} \dots \boldsymbol{q}_{l_k} A \emptyset^*.$$

If $1 < l_1 \leq \dots \leq l_k$, the former inequivalence is witnessed by the instance $XA^{l_1} BA^{l_2} B \dots A^{l_k} BA^{l_k} BX$. Less generally, $XA^* \boldsymbol{q_2 q_3} X \emptyset^* \not\equiv X \bot^* \boldsymbol{q_2 q_3} X \emptyset^*$ can be shown by $XAABAAABAAABX$ or by $XAABABAABABX$ (spot the difference and see ($\mathfrak{n}_2$) below).

Here, we consider the class $\mathcal{P}^\sigma[\mathsf{U}]$ of *peerless* queries given by (5), in which, for any $i$, either $\lambda_i = \bot$ or the sets $\lambda_i$ and $\rho_i$ are *incomparable* with respect to $\subseteq$. Our main result is that $\mathcal{P}^\sigma[\mathsf{U}]$ is polynomially characterisable within $\mathcal{Q}^\sigma_p[\mathsf{U}]$.

We start with a general observation. Consider two queries $\boldsymbol{q} = \rho_0 \lambda^*_1 \dots \lambda^*_n \rho_n \emptyset^*$ and $\boldsymbol{q}' = \rho_0 \mu^*_1 \dots \mu^*_n \rho_n \emptyset^*$. We say that $\lambda_i \neq \bot$ *subsumes* $\mu_j \neq \bot$ if either $i = j$ and $\mu_j \subseteq \lambda_i$, or $j < i$ and $\mu_j \rho_j \dots \rho_{i-1} \Subset \rho_j \dots \rho_{i-1} \lambda_i$, or $j > i$ and $\rho_i \dots \rho_{j-1} \mu_j \Subset \lambda_i \rho_i \dots \rho_{j-1}$. In the last two cases,

$$\mu_j \subseteq \rho_j \subseteq \dots \subseteq \rho_{i-1} \subseteq \lambda_i, \;\; \mu_j \subseteq \rho_{j-1} \subseteq \dots \subseteq \rho_i \subseteq \lambda_i,$$

respectively. Note that, for peerless $\boldsymbol{q}$, the last inclusion is impossible. If $\lambda_i$ and $\mu_j$ subsume each other, in which case $\lambda_i = \mu_j$, we call $(\lambda_i, \mu_j)$ a *matching pair*. Observe also that, for $\mathcal{D}^i_{\boldsymbol{q}} = \rho_0 \dots \rho_{i-1} \lambda_i \rho_i \dots \rho_n$, if $\mathcal{D}^i_{\boldsymbol{q}} \models \boldsymbol{q}'$, then $\lambda_i$ subsumes some $\mu_j$: $\rho_0 \dots \rho_n \emptyset \Subset \mathcal{D}^i_{\boldsymbol{q}}$ means that $\lambda_i$ subsumes $\mu_{n+1} = \emptyset$, and $\rho_0 \dots \mu_j \dots \rho_n \Subset \mathcal{D}^i_{\boldsymbol{q}}$ that $\lambda_i$ subsumes $\mu_j$. The proof of the next lemma is given in the full version:

**Lemma 12.** *For any queries $\boldsymbol{q}$ and $\boldsymbol{q}'$ as above, either $(i)$ each $\lambda_i \neq \bot$ subsumes $\mu_j$ occurring in some matching pair $(\lambda_k, \mu_j)$ or $(ii)$ $\boldsymbol{q}$ and $\boldsymbol{q}'$ are separated by a data instance of the form $\mathcal{D}^i_{\boldsymbol{q}}$ or $\mathcal{D}^j_{\boldsymbol{q}'}$. Also, if $\boldsymbol{q}$ is peerless, $\lambda_i$ can only subsume $\mu_j$ in the matching pair $(\lambda_i, \mu_j)$ with $i \geq j$, in which case $\mu_j = \rho_j = \dots = \rho_{i-1} = \lambda_i$.*

Note that the number of data instances of the form $\mathcal{D}^i_{\boldsymbol{q}'}$ for all possible $\mathcal{Q}^\sigma_p[\mathsf{U}]$-queries $\boldsymbol{q}'$ can be exponential in $|\sigma|$. The following example indicates how to overcome this issue.

**Example 13.** Let $\sigma = \{A, B, C, D, X\}$. To separate the query $X\{C, D\}^*A\emptyset^*$ from any $X\lambda^*A\emptyset^*$ with $A, D \notin \lambda$, we can use $\mathcal{D} = X\sigma \setminus \{A, D\}A$.

**Theorem 14.** *The class $\mathcal{P}^\sigma[\mathsf{U}]$ is polynomially characterisable within $\mathcal{Q}_p^\sigma[\mathsf{U}]$.*

*Proof sketch.* We show that any $\boldsymbol{q} = \rho_0\lambda_1^*\rho_1\lambda_2^* \dots \lambda_n^*\rho_n\emptyset^*$ in $\mathcal{P}^\sigma[\mathsf{U}]$ is characterised by the example set $E = (E^+, E^-)$ where $E^+$ contains all data instances of the following forms:

($\mathfrak{p}_0$) $\rho_0 \dots \rho_n$,

($\mathfrak{p}_1$) $\rho_0 \dots \rho_{i-1}\lambda_i\rho_i \dots \rho_n = \mathcal{D}^i_{\boldsymbol{q}}$,

($\mathfrak{p}_2$) $\rho_0 \dots \rho_{i-1}\lambda_i^k\rho_i \dots \rho_{j-1}\lambda_j\rho_j \dots \rho_n = \mathcal{D}^j_{i,k}$, for $i < j$ and $k = 1, 2$;

and $E^-$ has all instances that are *not* in $\boldsymbol{L}(\boldsymbol{q})$ of the forms:

($\mathfrak{n}_0$) $\sigma^n$ and $\sigma^{n-i}\sigma \setminus \{A\}\sigma^i$, for $A \in \rho_i$,

($\mathfrak{n}_1$) $\rho_0 \dots \rho_{i-1}\sigma \setminus \{A, B\}\rho_i \dots \rho_n$, for $A \in \lambda_i \cup \{\bot\}$ and $B \in \rho_i \cup \{\bot\}$,

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

$$\mathcal{D}^i_A = \rho_0 \dots \rho_{i-1}(\sigma \setminus \{A\})\rho_i\lambda_{i+1}^{k_{i+1}} \dots \lambda_n^{k_n}\rho_n, \qquad (6)$$

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 $\lambda_j$, for all $j \leq i$, with $\bot$. Note that $\mathcal{D}^i_A \not\models \boldsymbol{q}$ for peerless $\boldsymbol{q}$.

By definition, $\boldsymbol{q}$ fits $E$ and $|E|$ is polynomial in $|\boldsymbol{q}|$. We prove in the full version that $E$ uniquely characterises $\boldsymbol{q}$. ❑

One reason why this construction does not generalise to the whole $\mathcal{Q}_p^\sigma[\mathsf{U}]$ is that $\mathcal{D}^i_A \not\models \boldsymbol{q}^\dagger$ does not imply $\mathcal{D}^i_A \not\models \boldsymbol{q}$ for non-peerless $\boldsymbol{q}$, as shown by the following example:

**Example 15.** Let $\boldsymbol{q} = XA^*AB^*A\bot^*AB^*AA^*BB^*X\emptyset^*$. For any data instance $\mathcal{D}^3_\bot$ satisfying (6)—for example, $\mathcal{D}^3_\bot = XAA\sigma ABABX$—we have $\mathcal{D}^3_\bot \models \boldsymbol{q}$.

## 7 Characterisability in $\mathcal{Q}[\Diamond]$

In the previous two sections, we have investigated characterisability of path-shaped queries. Here, we first justify that restriction by exhibiting two examples that show how temporal branching can destroy polynomial characterisability in $\mathcal{Q}[\Diamond]$. Both examples make use of *unbalanced* queries, in which different branches have different length. We then show that this is no accident: one can at least partially restore polynomial characterisability for classes without unbalanced queries.

We start by observing that, without loss of generality, it is enough to consider conjunctions of path queries only:

**Lemma 16.** *For every $\boldsymbol{q} \in \mathcal{Q}[\Diamond]$, one can compute in polynomial time an equivalent query of the form $\boldsymbol{q}_1 \wedge \dots \wedge \boldsymbol{q}_n$ with $\boldsymbol{q}_i \in \mathcal{Q}_p[\Diamond]$, for $1 \leq i \leq n$.*

The first example showing non-polynomial characterisability is rather straightforward but requires unbounded branching and an unbounded number of atoms. We write queries $\boldsymbol{q} \in \mathcal{Q}_p^\sigma[\Diamond]$ of the form

$$\boldsymbol{q} = \rho_0 \wedge \Diamond(\rho_1 \wedge \Diamond(\rho_2 \wedge \dots \wedge \Diamond\rho_n)) \qquad (7)$$

as words $\rho_0\rho_1 \dots \rho_n$ over $2^\sigma$ (omitting but not forgetting $\lambda_i^* = \emptyset^*$ from (5)) and also use $\rho_0\rho_1 \dots \rho_n$ to denote the data instance defined by $\boldsymbol{q}$.

**Example 17.** Consider $\boldsymbol{q}_n = \boldsymbol{s}_1 \wedge \dots \wedge \boldsymbol{s}_n$, where $n \geq 2$ and each $\boldsymbol{s}_i$ is a word repeating $n$ times the sequence $A_1 \dots A_n$ (of singletons) with omitted $A_i$. Now, consider the queries $\boldsymbol{q}_n^{\boldsymbol{p}} = \boldsymbol{q}_n \wedge \boldsymbol{p}$, where $\boldsymbol{p} = \Diamond(A_{i_1} \wedge \Diamond(A_{i_2} \wedge \dots \wedge \Diamond A_{i_n}))$ and $A_{i_1} \dots A_{i_n}$ is a permutation of $A_1 \dots A_n$. Then $\boldsymbol{q}_n^{\boldsymbol{p}} \models \boldsymbol{q}_n$ and $\boldsymbol{q}_n \not\models \boldsymbol{q}_n^{\boldsymbol{p}}$ as shown by the data instance $\boldsymbol{s}_{i_1}\boldsymbol{s}_{i_2} \dots \boldsymbol{s}_{i_n}$. Moreover, if $\mathcal{D} \models \boldsymbol{q}_n$, $\mathcal{D} \not\models \boldsymbol{p}$ and $\boldsymbol{p}' \neq \boldsymbol{p}$, then $\mathcal{D} \models \boldsymbol{p}'$. It follows that, in any $E = (E^+, E^-)$ uniquely characterising $\boldsymbol{q}_n$, the set $E^+$ contains at least $n!$ data instances.

The class $\mathcal{Q}_{\leq n}[\Diamond]$ of queries of *branching factor* at most $n$ contains all queries in $\mathcal{Q}[\Diamond]$ that are equivalent to a query of the form $\boldsymbol{q}_1 \wedge \dots \wedge \boldsymbol{q}_m$ with $m \leq n$ and $\boldsymbol{q}_i \in \mathcal{Q}_p[\Diamond]$. We next provide an example of non-polynomial characterisability that requires four atoms and bounded branching only.

**Example 18.** Let $\sigma = \{A_1, A_2, B_1, B_2\}$, $\boldsymbol{q}_1 = \emptyset(\boldsymbol{s}\sigma)^n\boldsymbol{s}$, and $\boldsymbol{q}_2 = \emptyset\sigma^{2n+1}$, where $\boldsymbol{s} = \{A_1, A_2\}\{B_1, B_2\}$. Consider the set $P$ of $2^{n+1}$-many queries of the form $\emptyset\boldsymbol{s}_1 \dots \boldsymbol{s}_{n+1}$ with $\boldsymbol{s}_i$ either $\{A_1\}\{A_2\}$ or $\{B_1\}\{B_2\}$. Then $\boldsymbol{q}_1 \wedge \boldsymbol{q}_2 \not\models \boldsymbol{q}$ for any $\boldsymbol{q} \in P$ and, for any $\mathcal{D}$ with $\mathcal{D} \models \boldsymbol{q}_1 \wedge \boldsymbol{q}_2$, there is at most one $\boldsymbol{q} \in P$ with $\mathcal{D} \not\models \boldsymbol{q}$ (the proof is rather involved). It follows that $\boldsymbol{q}_1 \wedge \boldsymbol{q}_2 \not\models \boldsymbol{q}_1 \wedge \boldsymbol{q}_2 \wedge \boldsymbol{q}$ for all $\boldsymbol{q} \in P$, but $2^{n+1}$ positive examples are needed to separate $\boldsymbol{q}_1 \wedge \boldsymbol{q}_2$ from all $\boldsymbol{q}_1 \wedge \boldsymbol{q}_2 \wedge \boldsymbol{q}$ with $\boldsymbol{q} \in P$.

We next identify polynomially characterisable classes of $\mathcal{Q}[\Diamond]$-queries, assuming as before that $\rho_n \neq \emptyset$ in any $\boldsymbol{q}$ of the form (1). We call a query $\boldsymbol{q}_1 \wedge \dots \wedge \boldsymbol{q}_n \in \mathcal{Q}[\Diamond]$ with $\boldsymbol{q}_1, \dots, \boldsymbol{q}_n \in \mathcal{Q}_p[\Diamond]$ *balanced* if all $\boldsymbol{q}_i$ have the same depth; further, we call it *simple* if, in each $\boldsymbol{q}_i$ given by (1), $|\rho_j| = 1$ for all $j$. Let $\mathcal{Q}_b[\Diamond]$ denote the class of queries in $\mathcal{Q}[\Diamond]$ that are equivalent to a balanced query.

**Theorem 19.** $(i)$ *The class of simple queries in $\mathcal{Q}_b[\Diamond]$ is polynomially characterisable within $\mathcal{Q}_b[\Diamond]$.*

$(ii)$ *For any $n$, the class $\mathcal{Q}_b[\Diamond] \cap \mathcal{Q}_{\leq n}[\Diamond]$ is polynomially characterisable.*

*Proof sketch.* Let $\boldsymbol{q} \in \mathcal{Q}_p^\sigma[\Diamond]$. We start with a lemma on the existence of polynomial-size $\sigma$-data instances $\mathcal{D}_{\boldsymbol{q},k}$ such that $\mathcal{D}_{\boldsymbol{q},k} \not\models \boldsymbol{q}$ and $\mathcal{D}_{\boldsymbol{q},k} \models \boldsymbol{q}'$ for all $\boldsymbol{q}' \in \mathcal{Q}_p^\sigma[\Diamond]$ with $\boldsymbol{q}' \not\models \boldsymbol{q}$ and $tdp(\boldsymbol{q}') \leq k$. Note that such $\mathcal{D}_{\boldsymbol{q},k}$ do not exist in general.

**Example 20.** Let $\boldsymbol{q} = A \wedge B$. Then $A \not\models \boldsymbol{q}$ and $B \not\models \boldsymbol{q}$ but there does not exist any $\mathcal{D}_{\boldsymbol{q},0}$ such that $\mathcal{D}_{\boldsymbol{q},0} \not\models \boldsymbol{q}$, $\mathcal{D}_{\boldsymbol{q},0} \models A$ and $\mathcal{D}_{\boldsymbol{q},0} \models B$.

In the following lemma, we therefore assume that $\boldsymbol{q}$ does not speak about the initial timepoint.

**Lemma 21.** *Let $\boldsymbol{q} \in \mathcal{Q}_p^\sigma[\Diamond]$ be of the form $\Diamond\boldsymbol{q}'$ and let $k > 0$. Then one can construct in polynomial time a $\sigma$-data instance $\mathcal{D}_{\boldsymbol{q},k}$ such that $\mathcal{D}_{\boldsymbol{q},k} \not\models \boldsymbol{q}$ and $\mathcal{D}_{\boldsymbol{q},k} \models \boldsymbol{q}'$ for all $\boldsymbol{q}' \in \mathcal{Q}_p^\sigma[\Diamond]$ with $\boldsymbol{q}' \not\models \boldsymbol{q}$ and $tdp(\boldsymbol{q}') \leq k$.*

*Proof.* Assuming that $\boldsymbol{q} = \Diamond(\rho_1 \wedge \Diamond(\rho_2 \wedge \dots \wedge \Diamond\rho_n))$ with $\rho_i = \{A^i_1, \dots, A^i_{n_i}\}$ for $i \geq 1$, we set

$$\mathcal{D}_{\boldsymbol{q},k} = \sigma\boldsymbol{s}_1^k\sigma \cdots \boldsymbol{s}_{n-1}^k\sigma\boldsymbol{s}_n^k,$$

where $\boldsymbol{s}_i = \sigma \setminus \{A^i_1\} \dots \sigma \setminus \{A^i_{n_i}\}$. One can show by induction that $\mathcal{D}_{\boldsymbol{q},k}$ is as required. ❑

Using Lemma 21, for any $\boldsymbol{q} \in \mathcal{Q}^\sigma[\Diamond]$, one can construct a polynomial-size set of negative examples as follows. Suppose $\boldsymbol{q} = \boldsymbol{q}_1 \land \dots \land \boldsymbol{q}_n \in \mathcal{Q}^\sigma[\Diamond]$ with

$$\boldsymbol{q}_i = \rho^i_0 \land \Diamond(\rho^i_1 \land \Diamond(\rho^i_2 \land \dots \land \Diamond\rho_{n^i_i})).$$

Let $\rho = \bigwedge_{i=1}^n \rho^i_0$ and let $\boldsymbol{q}_i^-$ be $\boldsymbol{q}_i$ without the conjunct $\rho^i_0$, so Lemma 21 is applicable to $\boldsymbol{q}_i^-$. Now let $E^-_{\boldsymbol{q},m}$ contain the $\sigma$-data instances $\mathcal{D}_{\boldsymbol{q}_i^-,m}$ and $\sigma \setminus \{A\}\sigma^m$ for all $A \in \rho$.

**Lemma 22.** $(i)$ *For any $\mathcal{D} \in E^-_{\boldsymbol{q},m}$, we have $\mathcal{D} \not\models \boldsymbol{q}$.*

$(ii)$ *For any $\boldsymbol{q}' \in \mathcal{Q}^\sigma[\Diamond]$ with $\boldsymbol{q}' \not\models \boldsymbol{q}$ and $tdp(\boldsymbol{q}') \le m$, there exists $\mathcal{D} \in E^-_{\boldsymbol{q},m}$ with $\mathcal{D} \models \boldsymbol{q}'$.*

It follows from Lemma 22 that non-polynomial characterisability of $\mathcal{Q}[\Diamond]$-queries can only be caused by the need for super-polynomially-many positive examples. We now discuss the construction of positive examples in the proof of Theorem 19 $(ii)$; part $(i)$ is dealt with in the full version. Let $\boldsymbol{q} = \boldsymbol{q}_1 \land \dots \land \boldsymbol{q}_m \in \mathcal{Q}^\sigma_b[\Diamond] \cap \mathcal{Q}^\sigma_{\le n}[\Diamond]$ with $m \le n$ and

$$\boldsymbol{q}_i = \rho^i_0 \land \Diamond(\rho^i_1 \land \Diamond(\rho^i_2 \land \dots \land \Diamond\rho^i_N)).$$

For any map $f\colon \{1,\dots,m\} \to \{1,\dots,N\}$, construct a $\sigma$-data instance $\mathcal{D}_f$ by inserting $\rho^i_{f(i)}$ into the data instance $\sigma^N$ in position $f(i)$. Let $E^+$ contain the data instance $\rho\sigma^N$ for $\rho = \bigcup_{i=1}^m \rho^i_0$ and all the data instances $\mathcal{D}_f$. One can show that $(E^+, E^-)$ characterises $\boldsymbol{q}$ in $\mathcal{Q}_b[\Diamond] \cap \mathcal{Q}_{\le n}[\Diamond]$. ❑

## 8 2D Temporal Instance Queries

Now we consider 'two-dimensional' query languages that combine instance queries (over the object domain) in the standard description logics $\mathcal{EL}$ and $\mathcal{ELI}$ (Baader et al. 2017) with the *LTL*-queries (over the temporal domain) considered above. Our aim is to understand how far the characterisability results of the previous sections can be generalised to the 2D case. A *relational signature* is a finite set $\Sigma \neq \emptyset$ of unary and binary predicate symbols. A $\Sigma$-*data instance* $\mathcal{A}$ is a finite set of *atoms* $A(a)$ and $P(a,b)$ with $A, P \in \Sigma$ and *individual names* $a, b$. Let $ind(\mathcal{A})$ be the set of individual names in $\mathcal{A}$. We assume that $P^-(a,b) \in \mathcal{A}$ iff $P(b,a) \in \mathcal{A}$, calling $P^-$ the *inverse* of $P$ (with $P^{--} = P$). Let $S := P \mid P^-$. *Temporal instance queries* are defined by the grammar

$$\boldsymbol{q} \;:=\; \top \mid \bot \mid A \mid \exists S.\boldsymbol{q} \mid \boldsymbol{q}_1 \land \boldsymbol{q}_2 \mid \boldsymbol{op}\,\boldsymbol{q} \mid \boldsymbol{q}_1 \cup \boldsymbol{q}_2,$$

where $\boldsymbol{op} \in \{\bigcirc, \Diamond, \Diamond_r\}$. Such queries without temporal operators are called $\mathcal{ELI}$-*queries*; those of them without inverses $P^-$ are $\mathcal{EL}$-*queries*. A *temporal* $\Sigma$-*data instance* $\mathcal{D}$ is a finite sequence $\mathcal{A}_0, \dots, \mathcal{A}_n$ of $\Sigma$-data instances. We set $ind(\mathcal{D}) = \bigcup_{i=1}^n ind(\mathcal{A}_i)$. For any $\ell \in \mathbb{N}$ and $a \in ind(\mathcal{D})$, the *truth-relation* $\mathcal{D}, a, \ell \models \boldsymbol{q}$ is defined by induction:

$\mathcal{D}, a, \ell \models A$ iff $A(a) \in \mathcal{A}_\ell$,

$\mathcal{D}, a, \ell \models \exists S.\boldsymbol{q}$ iff there is $b \in ind(\mathcal{A}_\ell)$ such that $S(a,b) \in \mathcal{A}_\ell$ and $\mathcal{D}, b, \ell \models \boldsymbol{q}$,

with the remaining clauses being obvious generalisations of the *LTL* ones. An *example set* is a pair $E = (E^+, E^-)$ with finite sets $E^+$ and $E^-$ of *pointed temporal data instances* $(\mathcal{D}, a)$ such that $a \in ind(\mathcal{D})$. We say that $\boldsymbol{q}$ *fits* $E$ if $\mathcal{D}^+, a^+, 0 \models \boldsymbol{q}$ and $\mathcal{D}^-, a^-, 0 \not\models \boldsymbol{q}$, for all $(\mathcal{D}^+, a^+) \in E^+$ and $(\mathcal{D}^-, a^-) \in E^-$. As before, $E$ *uniquely characterises* $\boldsymbol{q}$ if $\boldsymbol{q}$ fits it and every $\boldsymbol{q}'$ fitting $E$ is logically equivalent to $\boldsymbol{q}$.

We need the following result on the unique characterisability of $\mathcal{ELI}$-queries.

**Theorem 23** (ten Cate and Dalmau 2021)**.** *The class of $\mathcal{ELI}$-queries is polynomially characterisable.*

Theorem 24 is proved by constructing frontiers in the set of $\mathcal{ELI}$-queries partially ordered by entailment, where a set $\mathcal{F}$ of $\mathcal{ELI}$-queries is called a *frontier* of an $\mathcal{ELI}$-query $\boldsymbol{q}$ if the following hold:

- $\boldsymbol{q} \models \boldsymbol{q}'$ and $\boldsymbol{q}' \not\models \boldsymbol{q}$, for all $\boldsymbol{q}' \in \mathcal{F}$;
- if $\boldsymbol{q} \models \boldsymbol{q}''$ for some $\mathcal{ELI}$-query $\boldsymbol{q}''$, then $\boldsymbol{q}'' \models \boldsymbol{q}$ or there exists $\boldsymbol{q}' \in \mathcal{F}$ with $\boldsymbol{q}' \models \boldsymbol{q}''$.

**Theorem 24** (ten Cate and Dalmau 2021)**.** *A frontier $\mathcal{F}(\boldsymbol{q})$ of any $\mathcal{ELI}$-query $\boldsymbol{q}$ can be computed in polynomial time.*

Theorem 23 follows from Theorem 24. For any $\mathcal{ELI}$-query $\boldsymbol{q}$, we denote by $\hat{\boldsymbol{q}}$ the tree-shaped data instance defined by $\boldsymbol{q}$ with designated root $a$. Then $\boldsymbol{q}$ is characterised by $E$ with $E^+ = \{\hat{\boldsymbol{q}}\}$ and $E^- = \{\hat{\boldsymbol{r}} \mid \boldsymbol{r} \in \mathcal{F}(\boldsymbol{q})\}$.

For any unrestricted temporal query language $\mathcal{Q}[\Phi]$ and $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}\}$, we denote by $\mathcal{Q}[\Phi] \otimes \mathcal{L}$ the set of all temporal instance queries with operators in $\Phi$ with (for $\mathcal{ELI}$) or without (for $\mathcal{EL}$) inverse predicates. We generalise the path-shaped queries $\mathcal{Q}_p[\Phi]$ as follows: $\mathcal{Q}_p[\Phi] \otimes \mathcal{L}$ denotes the class of queries $\boldsymbol{q}$ in $\mathcal{Q}[\Phi] \otimes \mathcal{L}$ such that, for any subquery $\boldsymbol{q}_1 \land \boldsymbol{q}_2$ of $\boldsymbol{q}$, either $\boldsymbol{q}_1$ or $\boldsymbol{q}_2$ do not have an occurrence of any operator in $\Phi$ that is not in the scope of $\exists S$. To illustrate, $\exists S.\Diamond A \land \Diamond\exists S.A$ is in $\mathcal{Q}_p[\Phi] \otimes \mathcal{L}$, but $\Diamond A \land \Diamond\exists S.A$ is not. We make two observations about unique characterisability in these 'full' combinations.

**Theorem 25.** $(i)$ *$\mathcal{Q}[\bigcirc, \Diamond] \otimes \mathcal{EL}$ is uniquely characterisable.*

$(ii)$ *$\mathcal{Q}_p[\bigcirc] \otimes \mathcal{ELI}$ and $\mathcal{Q}_p[\Diamond] \otimes \mathcal{ELI}$ are polynomially characterisable.*

Here, $(i)$ is shown similarly to Theorem 4 (it remains open whether it can be extended to $\mathcal{Q}[\bigcirc, \Diamond] \otimes \mathcal{ELI}$); $(ii)$ is proved by generalising Theorem 24 to temporal data instances.

We now show that the application of the DL constructor $\exists P$ to temporal queries with both $\bigcirc$ and $\Diamond$ destroys polynomial characterisability. Denote by $\mathcal{EL}(\mathcal{Q}_p[\bigcirc, \Diamond])$ the class of queries in $\mathcal{Q}_p[\bigcirc, \Diamond] \otimes \mathcal{EL}$ that contain no $\exists P$ in the scope of a temporal operator.

**Theorem 26.** *$\mathcal{EL}(\mathcal{Q}_p[\bigcirc, \Diamond])$ is not polynomially characterisable.*

*Proof sketch.* Consider queries $\boldsymbol{q}_n = \exists P.\boldsymbol{q}^n_1 \land \dots \land \exists P.\boldsymbol{q}^n_n$, in which each $\boldsymbol{q}^n_i$ corresponds to the regular expression

$$\underbrace{BB\emptyset^*A}_{1} \dots \underbrace{BB\emptyset^*A}_{i-1} \underbrace{\emptyset^*B\emptyset^*A}_{i} \underbrace{BB\emptyset^*A}_{i+1} \dots \underbrace{BB\emptyset^*A}_{n} \emptyset^*$$

(with omitted $\bot^* = \varepsilon$ in $BB$). One can show that any unique characterisation of $\boldsymbol{q}_n$ contains at least $2^n$ positive examples to separate it from all queries $\boldsymbol{q}_n \land \exists P.\boldsymbol{s}$ with

$$\boldsymbol{s} = \boldsymbol{o}_1(B \land \Diamond(A \land \boldsymbol{o}_2(B \land \Diamond(A \land \dots \land \boldsymbol{o}_n(B \land \Diamond A) \dots)))),$$

where $\boldsymbol{o}_i$ is $\bigcirc$ or $\Diamond\bigcirc$ if $i > 1$, and blank or $\Diamond$ if $i = 1$. ❑

The situation changes drastically if we do not admit temporal operators in the scope of $\exists P$. We start by investigating the class $\mathcal{Q}_p[\bigcirc, \Diamond_r](\mathcal{ELI})$ of queries of the form

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

where the $\boldsymbol{r}_i$ are $\mathcal{ELI}$-queries and $\boldsymbol{o}_i \in \{\bigcirc, \Diamond_r\}$. We can generalise the CQ-representation, the normal form, and the notion of lone conjunct from $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ to $\mathcal{Q}_p[\bigcirc, \Diamond_r](\mathcal{ELI})$ in a straightforward way. To formulate conditions $(\mathbf{n1})$–$(\mathbf{n4})$, we replace the set inclusions '$\rho_i \subseteq \rho_j$' by entailment '$\boldsymbol{r}_i \models \boldsymbol{r}_j$'. For example, $(\mathbf{n4})$ becomes

**(n4′)** $\boldsymbol{r}_0^{i+1} \not\models \boldsymbol{r}_{k_i}^i$ if $i > 0$, $\boldsymbol{q}_i$ is primitive and $R_{i+1}$ is $\leq$.

The condition for lone conjuncts now requires that $\boldsymbol{r}$ is not equivalent to any $\boldsymbol{q}_1 \wedge \boldsymbol{q}_2$ with $\mathcal{ELI}$-queries $\boldsymbol{q}_1, \boldsymbol{q}_2$ such that $\boldsymbol{q}_i \not\models \boldsymbol{r}$ for $i = 1, 2$. Then one can show again that every $\mathcal{Q}_p[\bigcirc, \Diamond_r](\mathcal{ELI})$-query is equivalent to a query in normal form, which can be computed in polynomial time.

**Theorem 27.** *The statements of Theorem 8 $(i)$–$(iv)$ also hold if one replaces $\mathcal{Q}_p[\bigcirc, \Diamond_r]$ by $\mathcal{Q}_p[\bigcirc, \Diamond_r](\mathcal{ELI})$.*

The proof generalises the example set defined in Theorem 8 using the frontiers provided by Theorem 24 as a *black box*. Indeed, in the definition of examples, replace $\rho_i$ by $\hat{\boldsymbol{r}}_i$, the data instance corresponding to the $\mathcal{ELI}$-query $\boldsymbol{r}_i$, and replace '$\rho \setminus \{A\}$ for $A \in \rho$' by 'the data instance corresponding to a query in $\mathcal{F}(\boldsymbol{r})$'. We choose a single individual, $a$, as the root of these data instances. For example, item (a) becomes:

(a′) replacing some $\boldsymbol{r}_j^i$ by the data instance corresponding to a query in $\mathcal{F}(\boldsymbol{r}_j^i)$ or removing the whole $\boldsymbol{r}_j^i = \emptyset$ for $i \neq 0$ and $j \neq 0$ from some $\boldsymbol{q}_i$.

Next, consider the class $\mathcal{Q}_p[\mathsf{U}](\mathcal{L})$ of queries of the form

$$\boldsymbol{q} = \boldsymbol{r}_0 \wedge (\boldsymbol{l}_1 \mathsf{U} (\boldsymbol{r}_1 \wedge (\boldsymbol{l}_2 \mathsf{U} (\dots (\boldsymbol{l}_n \mathsf{U} \boldsymbol{r}_n) \dots)))), \quad (8)$$

where $\boldsymbol{r}_i$ is an $\mathcal{L}$-query and $\boldsymbol{l}_i$ is either an $\mathcal{L}$-query or $\perp$, for $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}\}$. For the same reason as in the 1D case, we fix a finite signature $\Sigma$ of predicate symbols. Denote by $\mathcal{L}(\Sigma)$ and $\mathcal{Q}_p[\mathsf{U}](\mathcal{L})$ the set of queries in $\mathcal{L}$ and $\mathcal{Q}_p^\Sigma[\mathsf{U}](\mathcal{L})$, respectively, with predicate symbols in $\Sigma$. Aiming to generalise Theorem 14, we again translate set-inclusion to entailment, so the *peerless queries* $\mathcal{P}^\Sigma[\mathsf{U}](\mathcal{L})$ take the form (8) such that either $\boldsymbol{l}_i = \perp$ or $\boldsymbol{l}_i \not\models \boldsymbol{r}_i$ and $\boldsymbol{r}_i \not\models \boldsymbol{l}_i$.

**Theorem 28.** *Let $\Sigma$ be a finite relational signature. Then $\mathcal{P}^\Sigma[\mathsf{U}](\mathcal{EL})$ is polynomially characterisable within $\mathcal{Q}_p^\Sigma[\mathsf{U}](\mathcal{EL})$, while $\mathcal{P}^\Sigma[\mathsf{U}](\mathcal{ELI})$ is exponentially, but not polynomially, characterisable within $\mathcal{Q}_p^\Sigma[\mathsf{U}](\mathcal{ELI})$.*

To prove Theorem 28, we generalise the example set from the proof of Theorem 14. The positive examples are straightforward: simply replace $\rho_i$ and $\lambda_i$ by the data instances corresponding to $\boldsymbol{r}_i$ and $\boldsymbol{l}_i$ (and choose a single root individual). For the negative examples, we have to generalise the construction of $\sigma$, $\sigma \setminus \{A\}$, and $\sigma \setminus \{A, B\}$. For $\sigma$, this is straightforward as its role can now be played by the $\Sigma$-data instance $\mathcal{A}_\Sigma = \{A(a), R(a,a) \mid A, R \in \Sigma\}$ for which $\mathcal{A}_\Sigma \models \boldsymbol{q}(a)$ for all $\boldsymbol{q} \in \mathcal{ELI}(\Sigma)$. For $\sigma \setminus \{A\}$ and $\sigma \setminus \{A, B\}$, we require *split partners* defined as follows. Let $Q$ be a finite set of $\mathcal{L}(\Sigma)$-queries. A set $\mathcal{S}(Q)$ of pointed $\Sigma$-data instances $(\mathcal{A}, a)$ is called a *split partner* of $Q$ in $\mathcal{L}(\Sigma)$ if the following conditions are equivalent for all $\mathcal{L}(\Sigma)$-queries $\boldsymbol{q}'$:

- $\mathcal{A} \models \boldsymbol{q}'(a)$ for some $(\mathcal{A}, a) \in \mathcal{S}(Q)$;
- $\boldsymbol{q}' \not\models \boldsymbol{q}$ for all $\boldsymbol{q} \in Q$.

**Example 29.** The split partner of $\{\boldsymbol{q} = A\}$ in $\mathcal{EL}(\Sigma)$ is the singleton set containing $(\mathcal{A}_\Sigma^{-A}, a)$ with $\mathcal{A}_\Sigma^{-A}$ defined as $\{B(a), R(a,b), R(b,b), B'(b) \mid B \in \Sigma \setminus \{A\}, R, B' \in \Sigma\}$.

**Theorem 30.** *Fix $n > 0$. For any set $Q$ of $\mathcal{EL}(\Sigma)$-queries with $|Q| \leq n$, one can compute in polynomial time a split partner $\mathcal{S}(Q)$ of $Q$ in $\mathcal{EL}(\Sigma)$. For $\mathcal{ELI}$, one can compute a split partner in exponential time, which is optimal as even for singleton sets $Q$ of $\mathcal{ELI}(\Sigma)$-queries, no polynomial-size split partner of $Q$ in $\mathcal{ELI}(\Sigma)$ exists in general.*

The proof, given in the full version, requires (as does $\mathcal{A}_\Sigma$) the construction of non-tree-shaped data instances. Our results for $\mathcal{ELI}$ are closely related to the study of generalised dualities for homomorphisms between relational structures (Foniok, Nesetril, and Tardif 2008; Nesetril and Tardif 2005) but use pointed relational structures. The construction of $\mathcal{S}(Q)$ for $\mathcal{ELI}$ is based on a construction first introduced in (Bienvenu et al. 2014).

We obtain the negative examples for $\boldsymbol{q}$ of the form (8) by taking the following pointed data instances $(\mathcal{D}, a)$ (assuming that split partners take the form $(\mathcal{A}, a)$ for a fixed $a$):

($\mathrm{n}_0'$) $(\mathcal{A}_\Sigma^n, a)$ and $(\mathcal{A}_\Sigma^{n-i} \mathcal{A} \mathcal{A}_\Sigma^i, a)$, for $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{r}_i\})$;

($\mathrm{n}_1'$) $(\mathcal{D}, a) = (\hat{\boldsymbol{r}}_0 \dots \hat{\boldsymbol{r}}_{i-1} \mathcal{A} \hat{\boldsymbol{r}}_i \dots \hat{\boldsymbol{r}}_n, a)$ with $\mathcal{D}, a, 0 \not\models \boldsymbol{q}$ and $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i, \boldsymbol{r}_i\}) \cup \mathcal{S}(\{\boldsymbol{l}_i\}) \cup \mathcal{S}(\{\boldsymbol{r}_i\}) \cup \{(\mathcal{A}_\Sigma, a)\}$;

($\mathrm{n}_2'$) for all $i$ and $(\mathcal{A}, a) \in \mathcal{S}(\{\boldsymbol{l}_i\}) \cup \{(\mathcal{A}_\Sigma, a)\}$, *some* data instance

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

if any, such that $\max(\mathcal{D}_\mathcal{A}^i) \leq (n+1)^2$ and $\mathcal{D}_\mathcal{A}^i, a, 0 \not\models \boldsymbol{q}^\dagger$ for $\boldsymbol{q}^\dagger$ obtained from $\boldsymbol{q}$ by replacing all $\boldsymbol{l}_j$, $j \leq i$, with $\perp$.

We illustrate the construction by generalising Example 2.

**Example 31.** For $\boldsymbol{q} = \bigcirc A$ and any relational signature $\Sigma \ni A$, we obtain, after removing redundant instances, that $E^+ = \{(\emptyset\{A(a)\}, a)\}$ and $E^- = \{(\mathcal{A}_\Sigma \mathcal{A}_\Sigma^{-A}\{A(a)\}, a)\}$ characterise $\boldsymbol{q}$ within $\mathcal{Q}_p^\Sigma[\mathsf{U}](\mathcal{EL})$.

We finally generalise Theorem 19 $(ii)$ (part $(i)$ is not interesting since simple queries do not generalise to any new class of $\mathcal{ELI}$-queries). Query classes such as $\mathcal{Q}[\Diamond](\mathcal{EL})$ are defined in the obvious way by replacing in $\mathcal{Q}[\Diamond]$-queries conjunctions of atoms by $\mathcal{EL}$-queries.

**Theorem 32.** *The class $\mathcal{Q}_b[\Diamond](\mathcal{EL}) \cap \mathcal{Q}_{\leq n}[\Diamond](\mathcal{EL})$ is polynomially characterisable for any $n < \omega$.*

Again the positive and negative examples are obtained from the 1D case by replacing $\sigma$ by $\mathcal{A}_\Sigma$ and $\sigma \setminus \{A\}$ by appropriate split partners.

## 9 Applications to Learning

We apply our results on unique characterisability to exact learnability of temporal instance queries. Given a class $\mathcal{Q}$ of such queries, we aim to identify a *target query* $\boldsymbol{q} \in \mathcal{Q}$ using queries to an oracle. The learner knows $\mathcal{Q}$ and the signature $\sigma$ ($\Sigma$ in the 2D case) of $\boldsymbol{q}$. We allow only one type of queries, called *membership queries*, in which the learner picks a $\sigma$-data instance $\mathcal{D}$ and asks the oracle whether $\mathcal{D} \models \boldsymbol{q}$ holds. (In the 2D case, the learner picks a pointed $\Sigma$-data instance $(\mathcal{D}, a)$ and asks whether $\mathcal{D}, a, 0 \models \boldsymbol{q}$ holds.) The oracle answers 'yes' or 'no' truthfully. The class $\mathcal{Q}$ is (*polynomial time*) *learnable with membership queries* if there exists an algorithm that halts for any $\boldsymbol{q} \in \mathcal{Q}$ and computes (in polynomial time in the size of $\boldsymbol{q}$ and $\sigma/\Sigma$), using membership queries, a query $\boldsymbol{q}' \in \mathcal{Q}$ that is equivalent to $\boldsymbol{q}$. By default, the learner does not know $|\boldsymbol{q}|$ in advance but reflecting Theorem 8 $(iii)$, we also consider the case when $|\boldsymbol{q}|$ is known (which is common in active learning).

Obviously, unique characterisability is a necessary condition for learnability with membership queries. Conversely, if there is an algorithm that computes, for every $\boldsymbol{q} \in \mathcal{Q}$, an example set that uniquely characterises $\boldsymbol{q}$ within $\mathcal{Q}^{sig(\boldsymbol{q})}$, then $\mathcal{Q}$ is learnable with membership queries: enumerate $\mathcal{Q}^{sig(\boldsymbol{q})}$ starting with the smallest query $\boldsymbol{q}$, compute a characterising set $E$ for $\boldsymbol{q}$ and check using membership queries whether $\boldsymbol{q}$ is equivalent to the target query. Eventually the algorithm will terminate with a query that is equivalent to the target query. As all of our positive results on unique characterisability provide algorithms computing example sets, we directly obtain learnability with membership queries. Moreover, if the example sets are computed in exponential time, then we obtain an exponential-time learning algorithm: in the enumeration above only $|sig(\boldsymbol{q})|^{|\boldsymbol{q}|}$ queries are checked before the target query is found. Unfortunately, we cannot infer polynomial-time learnability from polynomial characterisability in this way.

A detailed analysis of polynomial-time learnability using membership queries is beyond the scope of this paper. Instead, we focus on one main result, the polynomial-time learnability of $\mathcal{Q}_p[\bigcirc, \Diamond](\mathcal{ELI})$.

**Theorem 33.** $(i)$ *The class of safe queries in* $\mathcal{Q}_p[\bigcirc, \Diamond_r](\mathcal{ELI})$ *is polynomial-time learnable with membership queries.*

$(ii)$ *The class* $\mathcal{Q}_p[\bigcirc, \Diamond_r](\mathcal{ELI})$ *is polynomial-time learnable with membership queries if the learner knows the size of the target query in advance.*

$(iii)$ *The class* $\mathcal{Q}_p[\bigcirc, \Diamond](\mathcal{ELI})$ *is polynomially-time learnable with membership queries.*

*Proof sketch.* We consider the 1D case without $\mathcal{ELI}$-queries first. $(i)$ Our proof strategy is to construct a query $\boldsymbol{q}'$ that agrees with $\boldsymbol{q}$ on the positive and negative examples for $\boldsymbol{q}'$ from Theorem 8. The algorithm proceeds by computing a data instance $\mathcal{D}$. Our aim is to arrive at $\mathcal{D}_b$ through iterations of steps, from which the required query can be 'read off'.

**Step 1.** First, identify the number of $\bigcirc$ and $\Diamond$ in $\boldsymbol{q}$ by asking membership queries of the form $\sigma^k$ incrementally, starting from $k = 1$, and then set $b = \min\{k \mid \sigma^k \models \boldsymbol{q}\} + 1$ and $\mathcal{D}_0 = \sigma^b$. Initialise $\mathcal{D} = \mathcal{D}_0$.

**Step 2.** Suppose that a data instance $\mathcal{D}'$ is obtained from $\mathcal{D}$ by applying one of the rules $(a)$–$(e)$ of Theorem 8. If $\mathcal{D}' \models \boldsymbol{q}$ then replace $\mathcal{D}$ with $\mathcal{D}'$. Repeat as long as possible. One can show that the number of applications of each rule is bounded by a polynomial in $|\sigma|$ and the size of $\boldsymbol{q}$, and so **Step 2** finishes in polynomial time.

**Step 3.** Suppose $\mathcal{D}$ contains $\emptyset^b \rho_0^i \emptyset^b$ and $|\rho_0^i| \geq 2$. Since rule $(a)$ does not apply, every homomorphism $h\colon \boldsymbol{q} \to \mathcal{D}$ sends some $t_1, \dots, t_l$ to $\rho_0^i$, for $l \geq 1$. As $\boldsymbol{q}$ does not contain lone conjuncts, $\boldsymbol{q}$ contains singleton primitive blocks at positions $t_1, \dots, t_l$. Suppose $\rho_0^i = \{A_1, \dots, A_{|\rho_0^i|}\}$ and let $w = \{A_1\}\emptyset^b\{A_2\}\emptyset^b \dots \{A_{|\rho_0^i|}\}\emptyset^b$ (the order in which $A_1, \dots, A_{|\rho_0^i|}$, the elements of $\rho_0^i$, are enumerated does not matter, we fix any one). Let $\mathcal{D}_k^i$ be obtained from $\mathcal{D}$ by replacing $\emptyset^b \rho_0^i \emptyset^b$ with $\emptyset^b(w)^k$. Notice that, for $k = |\boldsymbol{q}|$, we have $\mathcal{D}_k^i \models \boldsymbol{q}$; however, the algorithm is not given this $k$. Instead, the algorithm incrementally iterates starting from $k = 1$ until $\mathcal{D}_k^i \models \boldsymbol{q}$. Since $k \leq |\boldsymbol{q}|$, this takes polynomially-many iterations. Let $\mathcal{D}'$ be obtained from $\mathcal{D}_k^i$ by removing primitive blocks as long as $\mathcal{D}' \models \boldsymbol{q}$. Notice that rules $(a)$–$(e)$ do not apply to $\mathcal{D}'$. Replace $\mathcal{D}$ with $\mathcal{D}'$. Repeat **Step 3** as long as possible. Since no new lone conjuncts are introduced, the process finishes after polynomially-many steps.

**Step 4.** At this point of computation, the algorithm has identified all blocks of $\boldsymbol{q}$ but not the sequences of $\Diamond$ and $\Diamond_r$ between them. They can be easily determined based on the positive and negative examples $\mathcal{D}_i$ and $\mathcal{D}_i^-$.

The proof of $(ii)$ is similar, with a modified **Step 3**. Finally, $(iii)$ is a consequence of $(ii)$ as the size of the query $\boldsymbol{q}$ does not exceed $n = |\sigma|b$.

We obtain a learning algorithm for $\mathcal{Q}_p[\bigcirc, \Diamond_r](\mathcal{ELI})$ by combining the learning algorithm above with the learning algorithm for $\mathcal{ELI}$-queries by ten Cate and Dalmau (2021) using the positive and negative examples given in Theorem 27. Note that the data instance $\mathcal{A}_\Sigma$ is now used instead of $\sigma$ and that one has to 'unfold' such non tree-shaped data instances into tree-shaped ones. ❑

## 10 Conclusions

In this paper, we have considered temporal instance queries with *LTL* operators and started investigating their unique (polynomial) characterisability and exact learnability using membership queries. We have obtained both positive and negative results, depending on the available temporal operators and the allowed interaction between the temporal and object dimensions in queries. The results indicate that finding complete classifications of 1D and 2D temporal queries according to (polynomial) characterisability and learnability could be a very difficult task. In particular, interesting open problems include the polynomial characterisability of full $\mathcal{Q}_p^\sigma[\mathsf{U}]$, more general criteria of polynomial characterisability for temporal branching queries and other temporal operators, and the polynomial-time learnability of $\mathcal{Q}_p^\sigma[\mathsf{U}]$ and 2D extensions. From a conceptual viewpoint, it would be of interest to develop a framework that spells out explicitly the conditions that non-temporal queries should satisfy so that their combination with *LTL*-queries preserves polynomial characterisability and polynomial-time learnability.

## Acknowledgments

This research was supported by the EPSRC UK grants EP/S032207 and EP/S032282 for the joint project 'quant$^{MD}$: Ontology-Based Management for Many-Dimensional Quantitative Data'.

## References


Aarts, F., and Vaandrager, F. 2010. Learning i/o automata. In *International Conference on Concurrency Theory*, 71–85. Springer.

Alexe, B.; ten Cate, B.; Kolaitis, P. G.; and Tan, W. C. 2011. Characterizing schema mappings via data examples. *ACM Trans. Database Syst.* 36(4):23.

Angluin, D.; Frazier, M.; and Pitt, L. 1992. Learning conjunctions of Horn clauses. *Mach. Learn.* 9:147–164.

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

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

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

Artale, A.; Kontchakov, R.; Kovtunova, A.; Ryzhikov, V.; Wolter, F.; and Zakharyaschev, M. 2017. Ontology-mediated query answering over temporal data: A survey. In *Proc. of TIME 2017*, volume 90 of *LIPIcs*, 1:1–1:37. Schloss Dagstuhl, Leibniz-Zentrum für Informatik.

Baader, F.; Horrocks, I.; Lutz, C.; and Sattler, U. 2017. *An Introduction to Description Logics*. Cambride University Press.

Baader, F.; Borgwardt, S.; and Lippmann, M. 2015. Temporal query entailment in the description logic SHQ. *J. Web Semant.* 33:71–93.

Barceló, P., and Romero, M. 2017. The complexity of reverse engineering problems for conjunctive queries. In *Proc. of ICDT*, 7:1–7:17.

Bienvenu, M.; ten Cate, B.; Lutz, C.; and Wolter, F. 2014. Ontology-based data access: A study through disjunctive datalog, CSP, and MMSNP. *ACM Trans. Database Syst.* 39(4):33:1–33:44.

Borgwardt, S., and Thost, V. 2015. Temporal query answering in the description logic EL. In *Proc. of IJCAI 2015*, 2819–2825. AAAI Press.

Camacho, A., and McIlraith, S. A. 2019. Learning interpretable models expressed in linear temporal logic. In *Proc. of ICAPS 2018*, 621–630. AAAI Press.

Cassel, S.; Howar, F.; Jonsson, B.; and Steffen, B. 2016. Active learning for extended finite state machines. *Formal Aspects Comput.* 28(2):233–263.

Chomicki, J., and Toman, D. 2018. *Temporal Logic in Database Query Languages*. Springer. 3992–3998.

Fijalkow, N., and Lagarde, G. 2021. The complexity of learning linear temporal formulas from examples. *CoRR* abs/2102.00876.

Foniok, J.; Nesetril, J.; and Tardif, C. 2008. Generalised dualities and maximal finite antichains in the homomorphism order of relational structures. *Eur. J. Comb.* 29(4):881–899.

Fortin, M.; Konev, B.; Ryzhikov, V.; Savateev, Y.; Wolter, F.; and Zakharyaschev, M. 2022a. Reverse engineering of temporal queries with and without LTL ontologies: First steps (extended abstract). In *Proc. of DL'22*.

Fortin, M.; Konev, B.; Ryzhikov, V.; Savateev, Y.; Wolter, F.; and Zakharyaschev, M. 2022b. Unique Characterisability and Learnability of Temporal Instance Queries. *CoRR abs/2205.01651*.

Funk, M.; Jung, J. C.; Lutz, C.; Pulcini, H.; and Wolter, F. 2019. Learning description logic concepts: When can positive and negative examples be separated? In *Proc. of IJCAI*, 1682–1688.

Funk, M.; Jung, J. C.; and Lutz, C. 2021. Actively learning concepts and conjunctive queries under ELr-ontologies. In *Proc. of IJCAI 2021*, 1887–1893. ijcai.org.

Gutiérrez-Basulto, V.; Jung, J. C.; and Sabellek, L. 2018. Reverse engineering queries in ontology-enriched systems: The case of expressive Horn description logic ontologies. In *Proc. of IJCAI-ECAI*.

Hodkinson, I. M.; Wolter, F.; and Zakharyaschev, M. 2000. Decidable fragment of first-order temporal logics. *Ann. Pure Appl. Log.* 106(1-3):85–134.

Howar, F., and Steffen, B. 2018. 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 *LNCS*, 123–148. Springer.

Jung, J. C.; Lutz, C.; Pulcini, H.; and Wolter, F. 2020. Logical separability of incomplete data under ontologies. In *Proc. of KR 2020*, 517–528.

Kolaitis, P. G. 2011. Schema Mappings and Data Examples: Deriving Syntax from Semantics (Invited Talk). In *Proc. of FSTTCS 2011*, volume 13 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 25–25. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Konev, B.; Lutz, C.; Ozaki, A.; and Wolter, F. 2017. Exact learning of lightweight description logic ontologies. *J. Mach. Learn. Res.* 18:201:1–201:63.

Konev, B.; Ozaki, A.; and Wolter, F. 2016. A model for learning description logic ontologies based on exact learning. In *Proc. of AAAI*, 1008–1015. AAAI Press.

Lehmann, J., and Hitzler, P. 2010. Concept learning in description logics using refinement operators. *Machine Learning* 78:203–250.

Lemieux, C.; Park, D.; and Beschastnikh, I. 2015. General LTL specification mining (t). In *Proc. of ASE*, 81–92. IEEE.

Lutz, C.; Wolter, F.; and Zakharyaschev, M. 2008. Temporal description logics: A survey. In *Proc. of TIME 2008*, 3–14. IEEE Computer Society.

Martins, D. M. L. 2019. Reverse engineering database queries from examples: State-of-the-art, challenges, and research opportunities. *Inf. Syst.* 83:89–100.

Neider, D., and Gavran, I. 2018. Learning linear temporal properties. In *Proc. of FMCAD 2018*, 1–10. IEEE.

Nesetril, J., and Tardif, C. 2005. Short answers to exponentially long questions: Extremal aspects of homomorphism duality. *SIAM J. Discret. Math.* 19(4):914–920.

Schild, K. 1993. Combining terminological logics with tense logic. In *Proc. of EPIA'93*, volume 727 of *LNCS*, 105–120. Springer.

Shahbaz, M., and Groz, R. 2009. Inferring mealy machines. In *International Symposium on Formal Methods*, 207–222. Springer.

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

ten Cate, B.; Kolaitis, P. G.; Qian, K.; and Tan, W. 2018. Active learning of GAV schema mappings. In *Proc. of PODS 2018*, 355–368. ACM.

ten Cate, B.; Dalmau, V.; and Kolaitis, P. G. 2013. Learning schema mappings. *ACM Trans. Database Syst.* 38(4):28:1–28:31.

Tirtarasa, S., and Turhan, A.-Y. 2022. Computing Generalizations of Temporal $\mathcal{EL}$ Concepts with Next and Global. In *Proc. of SAC'22*. Association for Computing Machinery.