On queries with inequalities in $DL\text{-}Lite^{\neq}_{\mathcal{R}}$

Gianluca Cima[1], Federico Croce[1], Maurizio Lenzerini[1], Antonella Poggi[2,1], Elian Toccacieli[1]

[1]Dipartimento di Ingegneria Informatica, Automatica e Gestionale "Antonio Ruberti"
[2]Dipartimento di Lettere e Culture Moderne
Sapienza Università di Roma
lastname@diag.uniroma1.it

Abstract. It is well-known that answering conjunctive queries with inequalities ($CQ^{\neq}$s) over *DL-Lite*$_{\mathcal{R}}$ ontologies is in general undecidable. In this paper we consider the subclass of $CQ^{\neq}$s, called $CQ^{\neq,b}$s, where inequalities involve only distinguished variables or individuals. In particular, we tackle the problem of answering $CQ^{\neq,b}$s and unions thereof ($UCQ^{\neq,b}$s) over $DL\text{-}Lite^{\neq}_{\mathcal{R}}$ ontologies, where $DL\text{-}Lite^{\neq}_{\mathcal{R}}$ corresponds to *DL-Lite*$_{\mathcal{R}}$ without the Unique Name Assumption, and with the possibility of asserting inequalities between individuals, as in `OWL 2 QL`. As a first contribution, we show that answering $CQ^{\neq,b}$s over $DL\text{-}Lite^{\neq}_{\mathcal{R}}$ ontologies has the same computational complexity as the UCQ case over *DL-Lite*$_{\mathcal{R}}$, i.e., it is in AC^0 in data complexity, in PTIME in TBox complexity, and NP-complete in combined complexity. We then deal with the $UCQ^{\neq,b}$ case, and prove that answering $UCQ^{\neq,b}$s over $DL\text{-}Lite^{\neq}_{\mathcal{R}}$ ontologies is still in AC^0 in data complexity and in PTIME in TBox complexity, but is Π^p_2-hard in combined complexity.

1 Introduction

DL-Lite$_{\mathcal{R}}$ is the Description Logic (DL) of the *DL-Lite* family [6] which underpins the `OWL 2` profile `OWL 2 QL` [16]. It is arguably one of the most important formalisms of choice for representing ontologies in Ontology-based Data Access (OBDA) [18, 22] scenarios, where the aim is to access a typically huge amount of data residing in external data sources. In particular, *DL-Lite*$_{\mathcal{R}}$ has been designed so that answering unions of conjunctive queries (UCQs) can be reduced to evaluating first-order logic queries over the database storing the ABox assertions, and therefore is in AC^0 with respect to the size of the ABox, i.e., in the so-called *data complexity* [21].

While answering UCQs over *DL-Lite*$_{\mathcal{R}}$ ontologies has been extensively studied in recent years (e.g., by establishing bound on the size of rewritings [10], developing optimisation algorithms [19], and implementing systems for real-world applications [4, 5]), we argue that not much is known about the problem of answering conjunctive queries with inequalities ($CQ^{\neq}$s) and unions thereof ($UCQ^{\neq}$s). To the best of our knowledge, the basic facts that are known about these latter cases can be summarised as follows:

- In stark contrast to the UCQ case, answering $CQ^{\neq}$s over *DL-Lite*$_{\mathcal{R}}$ ontologies is in general undecidable [12].

– For subclasses of $\mathrm{CQ}^{\neq}$s and $\mathrm{UCQ}^{\neq}$s, named *local* $\mathrm{CQ}^{\neq}$s and *local* $\mathrm{UCQ}^{\neq}$s, respectively, query answering over *DL-Lite*$_{\mathcal{R}}$ ontologies is decidable, but with a high coNExpTime upper bound in data complexity. Furthermore, it is provably intractable (in general coNP-hard in data complexity) already for *local* $\mathrm{CQ}^{\neq}$s [12].
– For the subclass of $\mathrm{CQ}^{\neq}$ with bounded inequalities (called $\mathrm{CQ}^{\neq,b}$), where inequalities involve only individuals or distinguished variables, query answering over *DL-Lite*$_{\mathcal{R}}$ ontologies is in PTime in data complexity and in ExpTime in combined complexity [17].

Observe that all the above results hold regardless of whether the *Unique Name Assumption* (UNA) is enforced or not. Also, it is immediate to see that, for *DL-Lite*$_{\mathcal{R}}$, they do not provide the answer to the question whether answering $\mathrm{CQ}^{\neq,b}$s and unions thereof ($\mathrm{UCQ}^{\neq,b}$s) has the same complexity as the UCQ case.

As a first consideration on these classes of queries, we observe that, differently from the UCQ case [3], query answering over *DL-Lite*$_{\mathcal{R}}$ ontologies is sensitive to the adoption of the UNA, even for $\mathrm{CQ}^{\neq,b}$s, as shown in following example.

Example 1. Consider the *DL-Lite*$_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T} = \emptyset$ and $\mathcal{A} = \{P(a, b)\}$. For the $\mathrm{CQ}^{\neq,b}$ $q = \{(x, y) \mid P(x, y) \wedge x \neq y\}$, it is easy to see that the tuple $\langle a, b\rangle$ is in the certain answers of q over $\mathcal{O}$ under the UNA, while it is not if the UNA is not enforced. Indeed, for the model $\mathcal{M}$ of $\mathcal{O}$ with $a^{\mathcal{M}} = b^{\mathcal{M}} = e$ and $P^{\mathcal{M}} = \{(e, e)\}$, that does not respect the UNA, we have that $q^{\mathcal{M}} = \emptyset$. □

Notice, however, that answering $\mathrm{UCQ}^{\neq,b}$s over *DL-Lite*$_{\mathcal{R}}$ ontologies under the UNA is a straightforward generalisation of the UCQ case.

Proposition 1. *Answering $\mathrm{UCQ}^{\neq,b}$s over DL-Lite$_{\mathcal{R}}$ ontologies under the UNA is in AC^0 in data complexity, in* PTime *in TBox complexity, and NP-complete in combined complexity.*

Therefore, in what follows, we implicitly assume that the UNA is not enforced. In particular, in this paper we consider the DL *DL-Lite*$_{\mathcal{R}}^{\neq}$, which extends *DL-Lite*$_{\mathcal{R}}$ with the possibility of asserting inequalities between individuals, as in OWL 2 QL, and we present the following results:

– Answering $\mathrm{CQ}^{\neq,b}$s over *DL-Lite*$_{\mathcal{R}}^{\neq}$ ontologies has the same computational complexity of the UCQ case, i.e., it is in AC^0 in data complexity, in PTime in TBox complexity, and NP-complete in combined complexity (cf. Theorem 2).
– Answering $\mathrm{UCQ}^{\neq,b}$s over *DL-Lite*$_{\mathcal{R}}^{\neq}$ ontologies is Π_2^p-hard in combined complexity (cf. Theorem 3).
– Answering $\mathrm{UCQ}^{\neq,b}$s over *DL-Lite*$_{\mathcal{R}}^{\neq}$ ontologies is in AC^0 in data complexity, in PTime in TBox complexity, and in ExpTime in combined complexity (cf. Theorem 4).

Several recent works investigate the problem of answering UCQs over *DL-Lite*$_{\mathcal{R}}$ ontologies [3, 6, 13], and answering SPARQL queries over OWL 2 QL ontologies [2, 11, 14, 15]. However, none of them deal with queries containing inequalities. Conversely, inequality is considered in [7, 8, 12, 17]. As we said before, a crucial result in [12] shows that answering queries with inequalities over *DL-Lite*$_{\mathcal{R}}$ ontologies is in general undecidable. In [7, 8] the authors prove that answering UCQ$^{\neq}$s over OWL 2 QL ontologies under the *Direct Semantics Entailment Regime* [9] (i.e., the regime usually adopted for SPARQL queries) can be polynomially reduced to the evaluation of a Datalog program, and therefore is in PTIME in data complexity, and in EXPTIME in combined complexity. As already mentioned, in [17] the author shows that the same results hold also for CQ$^{\neq,b}$s under the standard semantics.

The paper is organized as follows. In Section 2 we provide some preliminaries on the languages considered in the paper. In Section 3 we illustrate the notion of chase that we use for *DL-Lite*$_{\mathcal{R}}^{\neq}$, and some related technical results. In Section 4 and Section 5 we present our results on CQ$^{\neq,b}$s and UCQ$^{\neq,b}$s, respectively. Finally, in Section 6 we conclude the paper with a discussion on future work.

2 Preliminaries

In this section, we first formally define the syntax and the semantics of *DL-Lite*$_{\mathcal{R}}^{\neq}$, and then we present the query languages considered in this paper.

Ontology language. Essentially, *DL-Lite*$_{\mathcal{R}}^{\neq}$ extends *DL-Lite*$_{\mathcal{R}}$ with the possibility of asserting inequalities between individuals. Formally, starting with an alphabet of *individuals*, *atomic concepts*, and *atomic roles*, that includes the binary relation symbol $\neq$, a *DL-Lite*$_{\mathcal{R}}^{\neq}$ *ontology*, or simply an ontology, is a pair $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, such that $\mathcal{T}$, called *TBox*, and $\mathcal{A}$, called *ABox*, are sets of axioms, that have, respectively, the following forms:

$\mathcal{T} : B_1 \sqsubseteq B_2$	$R_1 \sqsubseteq R_2$	(concept/role inclusion)
$B_1 \sqsubseteq \neg B_2$	$R_1 \sqsubseteq \neg R_2$	(concept/role disjointness)
$\mathcal{A} : A(a)$	$P(a, b)$	(concept/role membership)
	$a \neq b$	(inequality)

where a, b denote *individuals*, A and P denote an atomic concept and an atomic role, respectively, B_1, B_2 are *basic concepts*, i.e., expressions of the form A, $\exists P$, or $\exists P^-$, and R_1 and R_2 are *basic roles*, i.e., expressions of the form P, or P^-.

The semantics of a *DL-Lite*$_{\mathcal{R}}^{\neq}$ ontology $\mathcal{O}$ is specified through the notion of interpretation. An *interpretation* for $\mathcal{O}$ is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, where the *interpretation domain* $\Delta^{\mathcal{I}}$ is a non-empty, possibly infinite set of objects, and the *interpretation function* $\cdot^{\mathcal{I}}$ assigns to each individual a a domain object $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, to each atomic concept A a set of domain objects $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, to each atomic role a set of pairs of domain objects $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and to the special predicate "$\neq$" the set of all pairs of distinct domain objects, i.e., $\neq^{\mathcal{I}} = \{(o_1, o_2) \mid o_1, o_2 \in \Delta^{\mathcal{I}} \wedge o_1 \neq o_2\}$. The interpretation function extends to the other basic concepts and the other other basic roles as follows: (*i*) $(\exists P)^{\mathcal{I}} = \{o \mid \exists o'.(o, o') \in P^{\mathcal{I}}\}$, (*ii*) $(\exists P^-)^{\mathcal{I}} = \{o \mid \exists o'.(o', o) \in P^{\mathcal{I}}\}$, and (*iii*) $(P^-)^{\mathcal{I}} = \{(o, o') \mid (o', o) \in P^{\mathcal{I}}\}$.

An interpretation $\mathcal{I}$ *satisfies* a concept inclusion $B_1 \sqsubseteq B_2$ (respectively, role inclusion $R_1 \sqsubseteq R_2$) if $B_1^{\mathcal{I}} \subseteq B_2^{\mathcal{I}}$ (respectively, $R_1^{\mathcal{I}} \subseteq R_2^{\mathcal{I}}$), and it satisfies a concept disjointness $B_1 \sqsubseteq \neg B_2$ (respectively, role disjointness $R_1 \sqsubseteq \neg R_2$) if $B_1^{\mathcal{I}} \cap B_2^{\mathcal{I}} = \emptyset$ (respectively, $R_1^{\mathcal{I}} \cap R_2^{\mathcal{I}} = \emptyset$). An interpretation $\mathcal{I}$ satisfies a *DL-Lite*$_{\mathcal{R}}$ TBox $\mathcal{T}$ if it satisfies every axiom in $\mathcal{T}$. An interpretation $\mathcal{I}$ satisfies a *DL-Lite*$_{\mathcal{R}}^{\neq}$ ABox $\mathcal{A}$ if (*i*) $a^{\mathcal{I}} \in A^{\mathcal{I}}$

for every $A(a) \in \mathcal{A}$, (*ii*) $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in P^{\mathcal{I}}$ for every $P(a,b) \in \mathcal{A}$, and (*iii*) $a^{\mathcal{I}} \neq^{\mathcal{I}} b^{\mathcal{I}}$ for every $a \neq b \in \mathcal{A}$. Finally, a *DL-Lite*$_{\mathcal{R}}^{\neq}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ is *satisfiable* if it has a model, where a *model* is an interpretation $\mathcal{I}$ for $\mathcal{O}$ that satisfies both the TBox $\mathcal{T}$ and the ABox $\mathcal{A}$.

Query language. A *conjunctive query with inequalities (CQ*$^{\neq}$*)* over a *DL-Lite*$_{\mathcal{R}}^{\neq}$ ontology $\mathcal{O}$ is an expression of the form $q = \{\boldsymbol{x} \mid \phi(\boldsymbol{x}, \boldsymbol{y})\}$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are tuples of *variables*, called *distinguished* and *existential* variables of q, respectively, and $\phi(\boldsymbol{x}, \boldsymbol{y})$, called the *body* of q, is a finite conjunction of *DL-Lite*$_{\mathcal{R}}^{\neq}$ ABox assertions with variables that can appear in predicate arguments, i.e., atoms of the form $A(t_1)$, $P(t_1, t_2)$, or $t_1 \neq t_2$, where each t_j is either an individual of $\mathcal{O}$, or a variable in $\boldsymbol{x}$ or $\boldsymbol{y}$. We impose that every variable in $\boldsymbol{x}$ or $\boldsymbol{y}$ appears in some atom of $\phi(\boldsymbol{x}, \boldsymbol{y})$. If $\boldsymbol{x}$ is empty, then the query is called *boolean*. A CQ$^{\neq}$ q without atoms of the form $x_1 \neq x_2$ in its body is called a conjunctive query (CQ). An intermediate class of queries that lies between CQs and CQ$^{\neq}$s is the class of *conjunctive queries with bound inequalities (CQ*$^{\neq,b}$*)*. Specifically, a CQ$^{\neq,b}$ $q = \{\boldsymbol{x} \mid \phi(\boldsymbol{x}, \boldsymbol{y})\}$ is a CQ$^{\neq}$ whose inequalities involve only individuals or distinguished variables, i.e., for every atom $z_1 \neq z_2$ appearing in $\phi(\boldsymbol{x}, \boldsymbol{y})$, both z_1 and z_2 are not in $\boldsymbol{y}$. An UCQ (resp., UCQ$^{\neq,b}$, UCQ$^{\neq}$) is a union of a finite set of CQs (resp., CQ$^{\neq,b}$, CQ$^{\neq}$) with same arity.

The set of *certain answers* of an UCQ$^{\neq}$ q over a *DL-Lite*$_{\mathcal{R}}^{\neq}$ ontology $\mathcal{O}$, denoted by $cert(q, \mathcal{O})$, is the set of n-tuples $\boldsymbol{t}$ of individuals such that $\boldsymbol{t}^{\mathcal{I}} \in q^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{O}$, where $\boldsymbol{t}^{\mathcal{I}} = \langle t_1^{\mathcal{I}}, \ldots, t_n^{\mathcal{I}} \rangle$ for $\boldsymbol{t} = \langle t_1, \ldots, t_n \rangle$, and $q^{\mathcal{I}}$ denotes the evaluation of q over $\mathcal{I}$ seen as a relational database [1]. When q is a boolean query, we write $\mathcal{O} \models q$ if $q^{\mathcal{I}} = \{\langle\rangle\}$ (i.e., q is *true* in $\mathcal{I}$, also denoted by $\mathcal{I} \models q$) for every model $\mathcal{I}$ of $\mathcal{O}$. Observe that, if $\mathcal{O}$ is unsatisfiable, then $cert(q, \mathcal{O})$ is trivially the set of all possible n-tuples of individuals, where n is the arity of q (*ex falso quodlibet*).

When we talk about the problem of answering a class of queries $\mathcal{Q}$ over a class of DL ontologies $\mathcal{L}$, in fact we implicitly refer to the following *decision problem* (also known as the *recognition problem*): Given a query q in the class $\mathcal{Q}$, an $\mathcal{L}$-ontology $\mathcal{O}$, and an n-tuple of $\boldsymbol{t}$ of individuals of $\mathcal{O}$, check whether $\boldsymbol{t} \in cert(q, \mathcal{O})$.

***DL-Lite*$_{\mathcal{R}}$.** It is well-known (see e.g., [6]) that a *DL-Lite*$_{\mathcal{R}}$ ontology $\mathcal{O}$ is satisfiable if and only if $cert(\mathcal{V}_{\mathcal{O}}, \mathcal{O}^p) = \emptyset$, where $\mathcal{O}^p$ is obtained from $\mathcal{O}$ by removing the disjointness axioms, and $\mathcal{V}_{\mathcal{O}}$ is the *$\mathcal{O}$-violation query*, i.e., the boolean UCQ obtained by including a CQ of the form $\{() \mid A_1(x) \wedge A_2(x)\}$ (resp., $\{() \mid A_1(x) \wedge R(x,y)\}$, $\{() \mid R_1(x,y) \wedge R_2(x,z)\}$, and $\{() \mid R_1(x,y) \wedge R_2(x,y)\}$) for each disjointness axiom $A_1 \sqsubseteq \neg A_2$ (resp., $A_1 \sqsubseteq \neg \exists R$ or $\exists R \sqsubseteq \neg A_1$, $\exists R_1 \sqsubseteq \neg \exists R_2$, and $R_1 \sqsubseteq \neg R_2$), where an atom of the form $R(x,y)$ stands for either $P(x,y)$ if R denotes an atomic role P, or $P(y,x)$ if R denotes the inverse of an atomic role, i.e., $R = P^-$.

It is also well-known that if q is an UCQ over a satisfiable *DL-Lite*$_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, then $\mathsf{PerfectRef}(q, \mathcal{T})$ (where PerfectRef is the algorithm described in [6]) computes an UCQ whose evaluation over $db(\mathcal{A})$ (i.e., the ABox $\mathcal{A}$ seen as a relational database) returns exactly $cert(q, \mathcal{O})$, that is, $(\mathsf{PerfectRef}(q, \mathcal{T}))^{db(\mathcal{A})} = cert(q, \mathcal{O})$. Note that the algorithm PerfectRef ignores the disjointness axioms in $\mathcal{O}$.

3 The chase for *DL-Lite*$^{\neq}_{\mathcal{R}}$

The conceptual tool that we use for addressing the problem of answering UCQ$^{\neq,b}$s over *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontologies is a modification of the chase used for *DL-Lite*$_{\mathcal{R}}$ [6]. Specifically, given a *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, we build a (possibly infinite) structure, starting from $Chase^0(\mathcal{O}) = \mathcal{A}$, and repeatedly computing $Chase^{j+1}(\mathcal{O})$ from $Chase^j(\mathcal{O})$ by applying suitable rules, where each rule can be applied only if certain conditions hold. In doing so, we make use of a new infinite alphabet V of variables for introducing fresh unknown individuals, and we follow a deterministic strategy that is fair, i.e., it is such that if at some point a rule is applicable then it will be eventually applied. Finally, we set $Chase(\mathcal{O}) = \bigcup_{i \in \mathbb{N}} Chase^i(\mathcal{O})$. Observe that we make use of the additional binary predicate symbol ineq, which is used to record all inequalities logically implied by $\mathcal{O}$.

The rules we use include all the ones illustrated in [6]. For example, if $A_1 \sqsubseteq \exists P \in \mathcal{T}$, $A_1(a)$ is in $Chase^j(\mathcal{O})$, and there does not exist any b such that $P(a,b)$ is in $Chase^j(\mathcal{O})$, then we set $Chase^{j+1}(\mathcal{O}) = Chase^j(\mathcal{O}) \cup \{P(a,s)\}$, where $s \in V$ does not appear in $Chase^j(\mathcal{O})$. There are, however, crucial additions related to the ineq predicate. In what follows, when we say that $B(a)$ is in $Chase^j(\mathcal{O})$, where B is a basic concept, we mean $A(a) \in Chase^j(\mathcal{O})$ if $B = A$, $P(a,b) \in Chase^j(\mathcal{O})$ for some b, if $B = \exists P$, or $P(b,a) \in Chase^j(\mathcal{O})$ for some b, if $B = \exists P^-$. Also, when we say $R(a,b)$ is in $Chase^j(\mathcal{O})$, where R is a basic role, we mean $P(a,b) \in Chase^j(\mathcal{O})$, if $R = P$, or $P(b,a) \in Chase^j(\mathcal{O})$, if $R = P^-$. The additional rules are as follows:

- If $a \neq b$ is in $Chase^j(\mathcal{O})$, and $\mathsf{ineq}(a,b)$ is not in $Chase^j(\mathcal{O})$, then $Chase^{j+1}(\mathcal{O}) = Chase^j(\mathcal{O}) \cup \{\mathsf{ineq}(a,b)\}$;
- If $\mathsf{ineq}(a,b)$ is in $Chase^j(\mathcal{O})$, and $\mathsf{ineq}(b,a)$ is not in $Chase^j(\mathcal{O})$, then $Chase^{j+1}(\mathcal{O}) = Chase^j(\mathcal{O}) \cup \{\mathsf{ineq}(b,a)\}$;
- if $B_1 \sqsubseteq \neg B_2 \in \mathcal{T}$, $B_1(a), B_2(b)$ are in $Chase^j(\mathcal{O})$, and $\mathsf{ineq}(a,b)$ is not in $Chase^j(\mathcal{O})$, then $Chase^{j+1}(\mathcal{O}) = Chase^j(\mathcal{O}) \cup \{\mathsf{ineq}(a,b)\}$;
- if $R_1 \sqsubseteq \neg R_2 \in \mathcal{T}$, $R_1(c,a), R_2(c,b)$ are in $Chase^j(\mathcal{O})$, and $\mathsf{ineq}(a,b)$ is not in $Chase^j(\mathcal{O})$ then $Chase^{j+1}(\mathcal{O}) = Chase^j(\mathcal{O}) \cup \{\mathsf{ineq}(a,b)\}$.

From $Chase(\mathcal{O})$ it is immediate to define an interpretation $\mathcal{I}_{\mathcal{O}}$ for $\mathcal{O}$, extended in order to deal with predicate ineq, as follows:

- $\Delta^{\mathcal{I}_{\mathcal{O}}} = V_{\mathcal{O}} \cup V$, where $V_{\mathcal{O}}$ is the set of individuals occurring in $\mathcal{O}$;
- $e^{\mathcal{I}_{\mathcal{O}}} = e$ for every individual $o \in \Delta^{\mathcal{I}_{\mathcal{O}}}$;
- $A^{\mathcal{I}_{\mathcal{O}}} = \{e \mid A(e)$ occurs in $Chase(\mathcal{O})\}$ for every atomic concept A;
- $P^{\mathcal{I}_{\mathcal{O}}} = \{(e_1, e_2) \mid P(e_1, e_2)$ occurs in $Chase(\mathcal{O})\}$ for every atomic role P;
- $\mathsf{ineq}^{\mathcal{I}_{\mathcal{O}}} = \{(e_1, e_2) \mid \mathsf{ineq}(e_1, e_2)$ occurs in $Chase(\mathcal{O})\}$.

Note that, by definition, $\neq^{\mathcal{I}_{\mathcal{O}}}$ is the set of all pairs of distinct individuals in $(V_{\mathcal{O}} \cup V)$, i.e. $\neq^{\mathcal{I}_{\mathcal{O}}} = \{(e_1, e_2) \mid e_1, e_2 \in (V_{\mathcal{O}} \cup V) \wedge e_1 \neq e_2\}$.

The next proposition shows that the interpretation $\mathcal{I}_{\mathcal{O}}$ plays a crucial role in *DL-Lite*$^{\neq}_{\mathcal{R}}$.

Proposition 2. *If $\mathcal{M} = \langle \Delta^{\mathcal{M}}, \cdot^{\mathcal{M}} \rangle$ is a model of a DL-Lite$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O}$, then there exists a function Ψ from $\Delta^{\mathcal{I}_{\mathcal{O}}} = V_{\mathcal{O}} \cup V$ to $\Delta^{\mathcal{M}}$ such that:*

1. *for every $e \in \Delta^{\mathcal{I}_{\mathcal{O}}}$, if $e \in A^{\mathcal{I}_{\mathcal{O}}}$, then $\Psi(e) \in A^{\mathcal{M}}$;*

2. *for every pair* $e_1, e_2 \in \Delta^{\mathcal{I}_\mathcal{O}}$*, if* $(e_1, e_2) \in P^{\mathcal{I}_\mathcal{O}}$*, then* $(\Psi(e_1), \Psi(e_2)) \in P^{\mathcal{M}}$*;*
3. *for every pair* $e_1, e_2 \in \Delta^{\mathcal{I}_\mathcal{O}}$*, if* $(e_1, e_2) \in \mathsf{ineq}^{\mathcal{I}_\mathcal{O}}$*, then* $\Psi(e_1) \neq \Psi(e_2)$*.*

Proof (Sketch). The proofs of 1. and 2. are similar to that of Lemma 28 of [6]. The proof of 3. is based on showing that the interpretation $\mathcal{I}_\mathcal{O}$ enjoys the following crucial property: for every pair of individuals $a, b \in V_\mathcal{O}$, $(a, b) \in \mathsf{ineq}^{\mathcal{I}_\mathcal{O}}$ if and only if for *every model* $\mathcal{M}$ of $\mathcal{O}$, $a^{\mathcal{M}} \neq b^{\mathcal{M}}$. □

The above proposition shows the role of predicate ineq, and the importance of distinguishing between $\neq$ and ineq. Indeed, since in $\mathcal{I}_\mathcal{O}$ two different elements e_1, e_2 satisfy $e_1 \neq e_2$, condition 3 in Proposition 2 does not hold with $\neq$ in place of ineq.

Note that if $\mathcal{I}_\mathcal{O}$ satisfies all the axioms of $\mathcal{O}$, then it is a model of $\mathcal{O}$, and therefore $\mathcal{O}$ is satisfiable. Otherwise, it can be seen that $\mathcal{I}_\mathcal{O}$ violates at least one disjointness or one inequality axiom of $\mathcal{O}$. Note in particular that $\mathcal{I}_\mathcal{O}$ violates a disjointness axiom if and only if $(\mathcal{V}_\mathcal{O})^{\mathcal{I}_\mathcal{O}} \neq \emptyset$, where $\mathcal{V}_\mathcal{O}$ is the $\mathcal{O}$-violation query (cf. Section 2). On the other hand, by construction, $\mathcal{I}_\mathcal{O}$ violates an inequality axiom if and only if there exists e in $(V_\mathcal{O} \cup V)$ such that $e \neq e$ occurs in $\mathcal{O}$. In both cases, by Proposition 2, there exists no interpretation that can be a model for $\mathcal{O}$, and hence $\mathcal{O}$ is unsatisfiable. Intuitively, this shows that similarly to the "canonical interpretation" of a *DL-Lite*$_\mathcal{R}$ ontology, $\mathcal{I}_\mathcal{O}$ is instrumental for checking the satisfiability of a *DL-Lite*$_\mathcal{R}^{\neq}$ ontology $\mathcal{O}$. Also, checking the satisfiability of a *DL-Lite*$_\mathcal{R}^{\neq}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ can be done in AC^0 in the size of $\mathcal{A}$ and in PTIME in the size of $\mathcal{T}$, exactly lime in *DL-Lite*$_\mathcal{R}$.

In what follows we implicitly assume to deal only with satisfiable *DL-Lite*$_\mathcal{R}^{\neq}$ ontologies. Also, we denote by $\delta(q)$ the query obtained by replacing each inequality atom $t_1 \neq t_2$ in q with the atom $\mathsf{ineq}(t_1, t_2)$. The next theorem states that $\mathcal{I}_\mathcal{O}$ is instrumental also for answering $CQ^{\neq,b}$s over *DL-Lite*$_\mathcal{R}^{\neq}$ ontologies.

Theorem 1. *If* $\mathcal{O}$ *is a DL-Lite*$_\mathcal{R}^{\neq}$ *ontology, and* q *is a CQ*$^{\neq,b}$ *over* $\mathcal{O}$*, then* $cert(q, \mathcal{O}) = \delta(q)^{\mathcal{I}_\mathcal{O}}$.

Proof (Sketch). If $\boldsymbol{t} \in \delta(q)^{\mathcal{I}_\mathcal{O}}$, then, based on Proposition 2, we can show that $\boldsymbol{t} \in q^{\mathcal{M}}$, for every model $\mathcal{M}$ of $\mathcal{O}$.

If $\boldsymbol{t} \notin \delta(q)^{\mathcal{I}_\mathcal{O}}$, and $\boldsymbol{t}$ does not satisfy in $\mathcal{I}_\mathcal{O}$ all atoms of $\delta(q)$ different from ineq atoms, then $\mathcal{I}_\mathcal{O}$ is itself a model of $\mathcal{O}$ showing that $\boldsymbol{t} \notin cert(q, \mathcal{O})$. On the other hand, if $\boldsymbol{t}$ satisfies in $\mathcal{I}_\mathcal{O}$ all atoms of $\delta(q)$ different from ineq atoms, then there is at least one atom of the form $\mathsf{ineq}(a, b)$ in $\delta(q)$ that is false in $\mathcal{I}_\mathcal{O}$. What we do in this case is to compute an interpretation $\mathcal{J}$ from $\mathcal{I}_\mathcal{O}$, where $a^{\mathcal{J}}$ and $b^{\mathcal{J}}$ coincide, and then we show that $\mathcal{J}$ is a model $\mathcal{M}$ of $\mathcal{O}$ such that $\boldsymbol{t} \notin q^{\mathcal{M}}$, thus showing that $\boldsymbol{t} \notin cert(q, \mathcal{O})$. □

4 Answering CQ$^{\neq,b}$s over *DL-Lite*$_\mathcal{R}^{\neq}$ ontologies

In this section, we study the problem of answering CQ$^{\neq,b}$s over *DL-Lite*$_\mathcal{R}^{\neq}$ ontologies. To this aim, we start by introducing some preliminary notation.

Given an inequality atom $x_1 \neq x_2$ and a disjointness axiom γ, $\rho(x_1 \neq x_2, \gamma)$, denotes the formula defined as follows:

– $\rho(x_1 \neq x_2, A_1 \sqsubseteq \neg A_2) = A_1(x_1) \wedge A_2(x_2)$,
– $\rho(x_1 \neq x_2, A \sqsubseteq \neg\exists R) = \rho(x_1 \neq x_2, \exists R \sqsubseteq \neg A) = A(x_1) \wedge R(x_2, z)$, where z is a fresh variable,
– $\rho(x_1 \neq x_2, \exists R_1 \sqsubseteq \neg\exists R_2) = R_1(x_1, z) \wedge R_2(x_2, w)$, where z and w are fresh variables, and
– $\rho(x_1 \neq x_2, R_1 \sqsubseteq \neg R_2) = R_1(x_1, z) \wedge R_2(x_2, z) \vee R_1(z, x_1) \wedge R_2(z, x_2)$.

where an atom of the form $R(x, y)$ stands for either $P(x, y)$ if R denotes an atomic role P, or $P(y, x)$ if R denotes the inverse of an atomic role, i.e., $R = P^-$.

Given an inequality atom $x_1 \neq x_2$ and a TBox $\mathcal{T}$ with disjointness axioms $\gamma_1, \ldots, \gamma_m$, we denote by $\sigma(x_1 \neq x_2, \mathcal{T})$ the disjunction

$$\mathsf{ineq}(x_1, x_2) \vee \mathsf{ineq}(x_2, x_1) \vee \bigvee_{\gamma_i \in \mathcal{T}} (\rho(x_1 \neq x_2, \gamma_i) \vee \rho(x_2 \neq x_1, \gamma_i))$$

Finally, we denote by $\tau(q, \mathcal{T})$ the query obtained from q by substituting every inequality $x_1 \neq x_2$ by $\sigma(x_1 \neq x_2, \mathcal{T})$, and then turning the resulting query into an equivalent union of CQs.

In Fig. 1, we present the algorithm $\mathsf{AnsCQ}^{\neq,b}(q, \mathcal{O})$ for computing the certain answers to a $\mathrm{CQ}^{\neq,b}$ q over a *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$.

Informally, the algorithm rewrites q into the UCQ $\tau(q, \mathcal{T})$, applies the algorithm PerfectRef described in [6] to $\tau(q, \mathcal{T})$, and then evaluates the resulting UCQ over the database $db^{\mathsf{ineq}}(\mathcal{A})$. Such database stores the object c (resp. the tuple c_1, c_2) in the table A (resp. R), for each assertion $A(c)$ (resp. $R(c_1, c_2)$) in $\mathcal{A}$, and stores the pair (c_1, c_2) in the table ineq for each assertion $c_1 \neq c_2$ in $\mathcal{A}$.

Algorithm $\mathsf{AnsCQ}^{\neq,b}(q, \mathcal{O})$
Input: n-ary $\mathrm{CQ}^{\neq,b}$ q, *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$
Output: a set of n-tuples of individuals of $\mathcal{O}$
begin
  $PR := \tau(q, \mathcal{T})$
  $PR := \mathsf{PerfectRef}(PR, \mathcal{T})$
  return $PR^{db^{\mathsf{ineq}}(\mathcal{A})}$
end

Fig. 1: The algorithm $\mathsf{AnsCQ}^{\neq,b}(q, \mathcal{O})$

Example 2. Consider the *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ with $\mathcal{T} = \{P_1 \sqsubseteq P_2, A_1 \sqsubseteq \neg A_2\}$, and the $\mathrm{CQ}^{\neq,b}$

$$q = \{(x_1, x_2) \mid P_2(x_1, x_2) \wedge x_1 \neq c\}$$

over $\mathcal{O}$. It is easy to see that $\sigma(x_1 \neq c, \mathcal{T})$ is the formula $\mathsf{ineq}(x_1, c) \vee \mathsf{ineq}(c, x_1) \vee A_1(x_1) \wedge A_2(c) \vee A_2(x_1) \wedge A_1(c)$ and $\tau(q, \mathcal{T}) = \{(x_1, x_2) \mid P_2(x_1, x_2) \wedge \mathsf{ineq}(x, c) \vee$

$P_2(x_1, x_2) \wedge \mathsf{ineq}(c, x_1) \vee P_2(x_1, x_2) \wedge A_1(x_1) \wedge A_2(c) \vee P_2(x_1, x_2) \wedge A_1(c) \wedge A_2(x)\}$. Finally, $\mathsf{PerfectRef}(\tau(q, \mathcal{T}), \mathcal{T})$ is $\{(x_1, x_2) \mid P_2(x_1, x_2) \wedge \mathsf{ineq}(x, c) \vee P_2(x_1, x_2) \wedge \mathsf{ineq}(c, x_1) \vee P_2(x_1, x_2) \wedge A_1(x_1) \wedge A_2(c) \vee P_2(x_1, x_2) \wedge A_1(c) \wedge A_2(x) \vee P_1(x_1, x_2) \wedge \mathsf{ineq}(x, c) \vee P_1(x_1, x_2) \wedge \mathsf{ineq}(c, x_1) \vee P_1(x_1, x_2) \wedge A_1(x_1) \wedge A_2(c) \vee P_1(x_1, x_2) \wedge A_1(c) \wedge A_2(x)$. □

Proposition 3. *If $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ is a DL-Lite$^{\neq}_{\mathcal{R}}$ ontology, and q is a CQ$^{\neq,b}$ over $\mathcal{O}$, then AnsCQ$^{\neq,b}(q, \mathcal{O})$ terminates and computes exactly $cert(q, \mathcal{O})$.*

Taking into account the computational complexity of algorithm $\mathsf{AnsCQ}^{\neq,b}$, the above proposition implies that answering CQ$^{\neq,b}$s over *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontologies has the same data and combined complexity as answering UCQs over *DL-Lite*$_{\mathcal{R}}$ ontologies.

Theorem 2. *Answering CQ$^{\neq}$s over DL-Lite$^{\neq}_{\mathcal{R}}$ is in AC^0 in data complexity, in* PTIME *in TBox complexity, and NP-complete in combined complexity.*

5 Answering UCQ$^{\neq,b}$s over *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontologies

In this section, we study the problem of answering UCQ$^{\neq,b}$s over *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontologies.

Observe that, differently from the UCQ case where for any UCQ $Q = q_1 \cup \ldots \cup q_n$ and any *DL-Lite*$_{\mathcal{R}}$ ontology $\mathcal{O}$ we have that $cert(Q, \mathcal{O}) = cert(q_1, \mathcal{O}) \cup \ldots \cup cert(q_n, \mathcal{O})$ [6], the next example shows that this is not the case if we consider UCQ$^{\neq,b}$s.

Example 3. Consider the *DL-Lite*$_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T} = \emptyset$ and $\mathcal{A} = \{P(a, b)\}$. For the boolean UCQ$^{\neq,b}$ $Q = q_1 \cup q_2$, where $q_1 = \{() \mid P(a, a)\}$ and $q_2 = \{() \mid a \neq b\}$, it is easy to see that $\mathcal{O} \models Q$. Indeed, for any model $\mathcal{M}$ of $\mathcal{O}$, if $a^{\mathcal{M}} = b^{\mathcal{M}}$, then $\mathcal{M} \models q_1$, otherwise $a^{\mathcal{M}} \neq b^{\mathcal{M}}$, then $\mathcal{M} \models q_2$. Notice, however, that both $\mathcal{O} \not\models q_1$ and $\mathcal{O} \not\models q_2$ hold. For the former, it is sufficient to simply consider a model $\mathcal{M}_1$ of $\mathcal{O}$ in which $a^{\mathcal{M}_1} \neq b^{\mathcal{M}_1}$. For the latter, it is sufficient to consider a model $\mathcal{M}_2$ of $\mathcal{O}$ in which $a^{\mathcal{M}_2} = b^{\mathcal{M}_2} = e$ and $P^{\mathcal{M}_2} = \{(e, e)\}$. □

The next theorem implies that, unless the polynomial hierarchy collapses to the first level, answering UCQ$^{\neq,b}$s over *DL-Lite*$_{\mathcal{R}}$ ontologies does not have the same combined complexity of the UCQ and CQ$^{\neq,b}$ cases.

Theorem 3. *Answering UCQ$^{\neq,b}$s over DL-Lite$_{\mathcal{R}}$ ontologies is Π^p_2-hard in combined complexity.*

Proof (Sketch). The proof of Π^p_2-hardness is by a LOGSPACE reduction from the $\forall\exists$-CNF problem, which is Π^p_2-complete [20]. □

In order to present our positive results for the UCQ$^{\neq,b}$ case, next we introduce the notion of e-satisfiability for an equivalence relation e. An equivalence relation e on a set of individuals $\mathcal{C}$ is a binary relation over $\mathcal{C}$ that is reflexive, symmetric, and transitive. In what follows, we write $c_1 \sim_e c_2$ to denote $(c_1, c_2) \in e$. Moreover, we denote by $\mathcal{O}_e = \langle \mathcal{T}, \mathcal{A}_e \rangle$ the *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology obtained from $\mathcal{O}$ by adding e to the signature of $\mathcal{O}$ as a new atomic role, and with $\mathcal{A}_e$ being the ABox obtained from $\mathcal{A}$ by adding the extension of the relation e, i.e., $\mathcal{A}_e = \mathcal{A} \cup e$.

Definition 1. *Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite$^{\neq}_{\mathcal{R}}$ ontology, e be an equivalence relation on a set $\mathcal{C}$ of individuals of $\mathcal{O}$, and $\mathcal{I}$ be a model of $\mathcal{O}$. Then, we say that $\mathcal{I}$ is an e-model of $\mathcal{O}$ if, for any pair of constants $c_1, c_2 \in \mathcal{C}$, we have that $c_1^{\mathcal{I}} = c_2^{\mathcal{I}}$ if and only if $c_1 \sim_e c_2$. Also, we say that $\mathcal{O}$ is e-satisfiable if it has an e-model.*

The next proposition states that checking for the e-satisfiability has the same computational complexity of checking the satisfiability.

Proposition 4. *Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite$^{\neq}_{\mathcal{R}}$ ontology, and let e be an equivalence relation on a set of constants $\mathcal{C}$ of $\mathcal{O}$. Checking whether $\mathcal{O}$ is e-satisfiable can be done in AC^0 in the size of $\mathcal{A}$ and in* PTIME *in the size of $\mathcal{T}$.*

Proof (Sketch). Let $\mathcal{V}^{\neq,e}_{\mathcal{O}}$ be the $\mathcal{O}_e$-violation query obtained by extending the boolean UCQ $\mathcal{V}_{\mathcal{O}}$ with the following CQs over the signature of $\mathcal{O}_e$:

- $\{() \mid A_1(x_1) \wedge A_2(x_2) \wedge e(x_1, x_2)\}$ for each axiom of the form $A_1 \sqsubseteq \neg A_2$,
- $\{() \mid A(x_1) \wedge R(x_2, y) \wedge e(x_1, x_2)\}$ for each axiom of the form $A \sqsubseteq \neg \exists R$ or of the form $\exists R \sqsubseteq \neg A$,
- $\{() \mid R_1(x_1, y) \wedge R_2(x_2, z) \wedge e(x_1, x_2)\}$ for each axiom of the form $\exists R_1 \sqsubseteq \neg \exists R_2$,
- $\{() \mid R_1(x_1, y_1) \wedge R_2(x_2, y_2) \wedge e(x_1, x_2) \wedge e(y_1, y_2)\}$ for each axiom of the form $R_1 \sqsubseteq \neg R_2$,

where an atom of the form $R(x, y)$ stands for either $P(x, y)$ if R denotes an atomic role P, or $P(y, x)$ if R denotes the inverse of an atomic role, i.e., $R = P^-$.

It can be readily seen that a *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O}$ is e-satisfiable if and only if $cert(\mathcal{V}^{\neq,e}_{\mathcal{O}}, \mathcal{O}^p_e) = \emptyset$ and there exists no $a \neq b$ occurring in $\mathcal{A}$ such that $a \sim_e b$.

Intuitively, for checking e-satisfiability we check whether the equivalence relation e contradicts a disjointness, or an inequality axiom. This also directly implies that checking whether $\mathcal{O}$ is e-satisfiable can be done by evaluating a suitable query over $db^{\mathsf{ineq}}(\mathcal{A})$, and therefore the problem is in AC^0 in the size of the ABox $\mathcal{A}$ and in PTIME in the size of the TBox $\mathcal{T}$, as required. □

Based on the above result, in Fig. 2 we provide the algorithm $\mathsf{AnsUCQ}^{\neq,b}(Q, \mathcal{O})$ for the problem of answering UCQ$^{\neq,b}$s over *DL-Lite*$_{\mathcal{R}}$ ontologies. Observe that it is enough to consider only boolean UCQ$^{\neq,b}$s. Indeed, given a UCQ$^{\neq,b}$ Q, a *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, and an n-tuple $\boldsymbol{t}$ of individuals of $\mathcal{O}$ of the same arity of Q, checking whether $\boldsymbol{t} \in cert(Q, \mathcal{O})$ is equivalent to checking whether $\mathcal{O} \models Q(\boldsymbol{t})$, where $Q(\boldsymbol{t})$ denotes the boolean UCQ$^{\neq,b}$ obtained by replacing appropriately the distinguished variables of each CQ q in Q with the individuals of $\boldsymbol{t}$.

Moreover, note that every boolean UCQ$^{\neq,b}$ Q is such that every inequality appearing in its body is of the form $a \neq b$, where both a and b are individuals of $\mathcal{O}$. In the algorithm, we denote by $ej(q)$ the function that, given a CQ q, returns the set of existential variables that appears more than two times in the body of q, i.e., the set of existential variables that are in join.

Intuitively, to say that $\mathcal{O} \not\models Q$, the algorithm seeks for a relation ψ between the individuals appearing in Q for which (*i*) the ontology $\mathcal{O}$ is e-satisfiable, where e is the equivalence relation induced by ψ, and (*ii*) the reformulated UCQ Q is not entailed by $\mathcal{O}_e$. In particular, Q is first reformulated by evaluating every inequality based on the

Algorithm $\mathsf{AnsUCQ}^{\neq,b}(Q,\mathcal{O})$
Input: a boolean UCQ$^{\neq,b}$ Q, and a *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontology $\mathcal{O} = \langle\mathcal{T},\mathcal{A}\rangle$
Output: true or **false**
begin
  let $\mathcal{C}_Q$ be the set of all individuals appearing in Q
  for each $\psi \subseteq (\mathcal{C}_Q \times \mathcal{C}_Q)$:
    let e be the reflexive, symmetric, and transitive closure of ψ
    if $\mathcal{O}$ is e-satisfiable **then**
      for each $q \in Q$ and inequality atom $c_1 \neq c_2 \in q$:
        if $c_1 \sim_e c_2$ **then**
          $q = q \setminus \{c_1 \neq c_2\}$
        else
          $Q = Q \setminus \{q\}$
      $Q' = Q$
      for each q in Q', $\mathcal{Y} \subseteq ej(q)$, and $y \in \mathcal{Y}$:
        let $y^1,\ldots,y^{m_y}$ denote the different occurrences of y in q
        replace each occurrence y^j of the variable y with a fresh existential variable z^j
        for each pair of newly introduced variables z^k, z^l with $k \neq l$:
          $q = q \cup \{e(z_i^k, z_i^l)\}$
        $Q = Q \cup q$
      for each q in Q and individual c in q:
        replace all the occurrences of c with a fresh existential variable y_c
        $q = q \cup \{e(y_c, c)\}$
      if $\mathcal{O}_e \not\models Q$ **then**
        return false
  return true
end

Fig. 2: The algorithm $\mathsf{AnsUCQ}^{\neq,b}(Q,\mathcal{O})$

equivalence relation e. Then, Q is further reformulated by allowing that in each CQ q of Q, and for some of the existential variables $y_1,\ldots,y_n$ appearing in q, different occurrences of each y_i may be mapped to distinct individuals of the set $\mathcal{C}_Q$, provided that these distinct individuals are in the same equivalence class of e. An analogous consideration is for the individuals appearing in the query, where the last step of the reformulation of Q allows that an existential variable y_c may match an individual c' that is not necessarily the individual c, but it is such that $c' \sim_e c$, i.e., it is in the same equivalence class of c.

Proposition 5. *If $\mathcal{O} = \langle\mathcal{T},\mathcal{A}\rangle$ is a DL-Lite$^{\neq}_{\mathcal{R}}$ ontology, and Q is a boolean UCQ$^{\neq,b}$ over $\mathcal{O}$, then $\mathsf{AnsUCQ}^{\neq,b}(q,\mathcal{O})$ terminates and returns **true** if and only if $\mathcal{O} \models Q$.*

With regard to the cost of the algorithm $\mathsf{AnsUCQ}^{\neq,b}(Q,\mathcal{O})$, observe that all the for-loops of the algorithm depend only on Q, and can be done in EXPTIME in its size. As for the e-satisfiability check, from Proposition 4, it can be done in AC^0 in the size of $\mathcal{A}$, and in PTIME in the size of $\mathcal{T}$. Also, since Q'_e is an UCQ, checking whether $\mathcal{O}_e \not\models Q'_e$ can be obviously done in AC^0 in the size of $\mathcal{A}$, in PTIME in the size of $\mathcal{T}$, and in ExpTime in the size of Q. From Proposition 5 and the above considerations, we easily get the following result.

Theorem 4. *Answering UCQ$^{\neq,b}$s over DL-Lite$^{\neq}_{\mathcal{R}}$ ontologies is in AC^0 in data complexity, in* PTime *in TBox complexity, and in* ExpTime *in combined complexity.*

6 Conclusion

In this paper we have singled out a specific class of queries, namely UCQ$^{\neq,b}$s, for which query answering over *DL-Lite*$^{\neq}_{\mathcal{R}}$ ontologies is still in AC^0 in data complexity and in PTime in TBox complexity. The algorithm is ExpTime in combined complexity, and we have shown that the problem is Π_2^p-hard.

There are several problems to consider for continuing the work presented here, the most obvious being trying to derive a matching Π_2^p upper bound in combined complexity of the above problem. Another interesting topic is to look for more subclasses of queries, or even more ontology languages for which answering queries with inequalities is decidable/tractable.

Acknowlegments

Work supported by MIUR under the SIR project "MODEUS" – grant n. RBSI14TQHQ, and by Sapienza under the research project "PRE-O-PRE".

References

1. S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison Wesley Publ. Co., 1995.
2. M. Arenas, G. Gottlob, and A. Pieris. Expressive languages for querying the semantic web. *ACM Trans. on Database Systems*, 43(3):13:1–13:45, 2018.
3. A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyaschev. The *DL-Lite* family and relations. *J. of Artificial Intelligence Research*, 36:1–69, 2009.
4. D. Calvanese, B. Cogrel, S. Komla-Ebri, R. Kontchakov, D. Lanti, M. Rezk, M. Rodriguez-Muro, and G. Xiao. Ontop: Answering SPARQL queries over relational databases. *Semantic Web J.*, 8(3):471–487, 2017.
5. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodriguez-Muro, R. Rosati, M. Ruzzi, and D. F. Savo. The Mastro system for ontology-based data access. *Semantic Web J.*, 2(1):43–53, 2011.
6. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Automated Reasoning*, 39(3):385–429, 2007.
7. G. Cima, G. De Giacomo, M. Lenzerini, and A. Poggi. On the SPARQL metamodeling semantics entailment regime for OWL 2 QL ontologies. In *Proc. of the 7th Int. Conf. on Web Intelligence, Mining and Semantics (WIMS)*, pages 10:1–10:6, 2017.
8. G. Cima, G. De Giacomo, M. Lenzerini, and A. Poggi. Querying OWL 2 QL under the SPARQL metamodeling semantics entailment regime. In *Proc. of the 25th Ital. Symp. on Advanced Database Systems (SEBD)*, volume 2037, page 165, 2017.

9. B. Glimm. Using SPARQL with RDFS and OWL entailment. In *Reasoning Web. Semantic Technologies for the Web of Data – 7th Int. Summer School Tutorial Lectures (RW)*, pages 137–201. 2011.
10. G. Gottlob, S. Kikot, R. Kontchakov, V. V. Podolskii, T. Schwentick, and M. Zakharyaschev. The price of query rewriting in ontology-based data access. *Artificial Intelligence*, 213:42–59, 2014.
11. G. Gottlob and A. Pieris. Beyond SPARQL under OWL 2 QL entailment regime: Rules to the rescue. In *Proc. of the 24th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 2999–3007, 2015.
12. V. Gutiérrez-Basulto, Y. A. Ibáñez-García, R. Kontchakov, and E. V. Kostylev. Queries with negation and inequalities over lightweight ontologies. *J. of Web Semantics*, 35:184–202, 2015.
13. S. Kikot, R. Kontchakov, and M. Zakharyaschev. Conjunctive query answering with OWL 2 QL. In *Proc. of the 13th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR)*, 2012.
14. R. Kontchakov, M. Rezk, M. Rodriguez-Muro, G. Xiao, and M. Zakharyaschev. Answering SPARQL queries over databases under OWL 2 QL entailment regime. In *Proc. of the 13th Int. Semantic Web Conf. (ISWC)*, pages 552–567, 2014.
15. M. Lenzerini, L. Lepore, and A. Poggi. A higher-order semantics for metaquerying in OWL 2 QL. In *Proc. of the 15th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR)*, pages 577–580, 2016.
16. B. Motik, B. Cuenca Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz. OWL 2 Web Ontology Language profiles (second edition). W3C Recommendation, World Wide Web Consortium, Dec. 2012. Available at `http://www.w3.org/TR/owl2-profiles/`.
17. A. Poggi. On the SPARQL direct semantics entailment regime for OWL 2 QL. In *Proc. of the 29th Int. Workshop on Description Logic (DL)*, 2016.
18. A. Poggi, D. Lembo, D. Calvanese, G. De Giacomo, M. Lenzerini, and R. Rosati. Linking data to ontologies. *J. on Data Semantics*, X:133–173, 2008.
19. R. Rosati and A. Almatelli. Improving query answering over *DL-Lite* ontologies. In *Proc. of the 12th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR)*, pages 290–300, 2010.
20. L. J. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):1–22, 1976.
21. M. Y. Vardi. The complexity of relational query languages. In *Proc. of the 14th ACM SIGACT Symp. on Theory of Computing (STOC)*, pages 137–146, 1982.
22. G. Xiao, D. Calvanese, R. Kontchakov, D. Lembo, A. Poggi, R. Rosati, and M. Zakharyaschev. Ontology-based data access: A survey. In *Proc. of the 27th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 5511–5519, 2018.