

The Complexity of Conjunctive Query Abduction in *DL-Lite*^{*}

Diego Calvanese¹, Magdalena Ortiz², Mantas Šimkus², and Giorgio Stefanoni^{1,2}

¹ KRDB Research Centre for Knowledge and Data

Free University of Bozen-Bolzano, Italy

calvanese@inf.unibz.it, giorgio.stefanoni@gmail.com

² Institute of Information Systems

Vienna University of Technology, Austria

ortiz@kr.tuwien.ac.at, simkus@dbai.tuwien.ac.at

Abstract. In order to meet usability requirements, most logic-based applications provide explanation facilities for reasoning services. This holds also for DLs, where research focused on the explanation of both TBox reasoning and, more recently, query answering. Besides explaining the presence of a tuple in a query answer, it is important to explain also why a given tuple is missing. We address this latter problem for (conjunctive) query answering over *DL-Lite* ontologies, by adopting abductive reasoning, that is, we look for additions to the ABox that force a given tuple to be in the result. As reasoning tasks, we consider existence and recognition of an explanation, and relevance and necessity of a certain assertion for an explanation. We characterize the computational complexity of these problems for subset minimal and cardinality minimal solutions.

1 Introduction

Query answering over ontologies formulated in Description Logics (DLs) has received considerable attention both in research and industry. Given an ontology, users typically pose queries over the conceptual schema and get answers that take into account the constraints specified at the conceptual level. Many efforts have concentrated on lightweight description logics. For instance, *DL-Lite_A* has been tailored for query answering over large data sets [7]. For this reason, expressive power is traded in favor of a better computational behavior in terms of data-complexity. In fact, conjunctive query answering in *DL-Lite_A* enjoys FOL-rewritability, i.e., it can be reduced to the problem of evaluating a suitably constructed FOL query over a database instance.

In order to meet usability requirements set by domain users, most logic-based applications provide explanation algorithms for reasoning services. This holds also for DLs, where research focused on the explanation of both TBox reasoning [12,6,14,3] and, more recently, query answering [5]. In addition, the latter paper advocates the importance of tackling the problem of explaining the absence of a tuple in the answers to a query over an ontology. This problem stems from the database community, where it has been solved in the context of databases extended with provenance information [9]. We address this problem by considering explanations for the absence of a

* This work was partially supported by the Austrian Science Fund (FWF) grant P20840.

tuple in the context of query answering over $DL\text{-Lite}_{\mathcal{A}}$ ontologies. We adopt *abductive reasoning* [10,11], that is, we consider which additions need to be made to the ABox to force the given tuple to be in the result. More precisely, given a TBox \mathcal{T} , an ABox \mathcal{A} , and a query q , an *explanation* for a given tuple \mathbf{t} is a new ABox \mathcal{U} such that the answer to q over $\langle \mathcal{T}, \mathcal{A} \cup \mathcal{U} \rangle$ contains \mathbf{t} . An important aspect in explanations is to provide the user with solutions that are simple to understand and free of redundancy, hence as small as possible. To address this requirement, we study various restrictions on solutions, in particular, we focus on subset minimal and cardinality minimal ones. We consider standard decision problems associated to logic-based abduction: (i) *existence* of an explanation, (ii) *recognition* of a given ABox as being an explanation, and (iii) *relevance* and (iv) *necessity* of an ABox assertion, i.e., whether it occurs in some or all explanations. After motivating such problems and formalizing them, we provide algorithms to solve them and a precise characterization of their computational complexity for $DL\text{-Lite}_{\mathcal{A}}$. The complexity results for the various reasoning tasks are summarized in Table 1.

2 Preliminaries

$DL\text{-Lite}_{\mathcal{A}}$. $DL\text{-Lite}_{\mathcal{A}}$ is a member of the $DL\text{-Lite}$ family of DLs [7], which have been designed for dealing efficiently with large amounts of extensional information. In $DL\text{-Lite}_{\mathcal{A}}$, concept expressions C , denoting sets of objects, and role expressions R , denoting binary relations between objects, are formed as follows:

$$C \longrightarrow A \mid \exists R, \quad R \longrightarrow P \mid P^-.$$

where A denotes an atomic concept and P an atomic role³. In a $DL\text{-Lite}_{\mathcal{A}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, the TBox \mathcal{T} consists of axioms of the form

$$\begin{array}{lll} C_1 \sqsubseteq C_2, & R_1 \sqsubseteq R_2, & (\text{funct } R), \\ C_1 \sqsubseteq \neg C_2, & R_1 \sqsubseteq \neg R_2, & \end{array}$$

and the ABox \mathcal{A} consists of assertions of the form $A(c)$ and $R(c, c')$, where c, c' are constants (or, *individuals*) from a countably infinite set \mathbf{C} . An interpretation is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, where $\Delta^{\mathcal{I}}$ is a non-empty *domain*, and the *interpretation function* $\cdot^{\mathcal{I}}$ is defined as usual. We adopt here the *unique name assumption* (UNA), i.e., $c_1^{\mathcal{I}} \neq c_2^{\mathcal{I}}$ for all $c_1, c_2 \in \mathbf{C}$ with $c_1 \neq c_2$. We refer to [7] for more details.

Conjunctive Queries. Let \mathbf{V} be a countably infinite set of variables. Expressions $A(t)$ and $P(t, t')$ are called *atoms*, where $t, t' \in \mathbf{V} \cup \mathbf{C}$. A *conjunctive query* (CQ) q is an expression $q(x_1, \dots, x_n) \leftarrow a_1, \dots, a_m$, where each a_i , $1 \leq i \leq m$, is an atom. Let $\mathbf{V}(q)$ denote the set of variables occurring in q , $\mathbf{C}(q)$ the set of constants in q , and let $at(q) = \bigcup_{1 \leq i \leq m} \{a_i\}$. A *match* for q in an interpretation \mathcal{I} is a mapping $\pi : \mathbf{V}(q) \cup \mathbf{C}(q) \rightarrow \Delta^{\mathcal{I}}$ such that π is the identity on constants, $\pi(t) \in A^{\mathcal{I}}$ for each

³ We ignore here the distinction between data values and objects, since it is immaterial for our results. As a consequence, we do not consider value domains and attributes, which are present in $DL\text{-Lite}_{\mathcal{A}}$.

$A(t) \in at(q)$, and $\langle \pi(t), \pi(t') \rangle \in P^{\mathcal{I}}$ for each $P(t, t') \in at(q)$. The tuple $\langle x_1, \dots, x_n \rangle$ is the tuple of *answer variables* of q . The *answer to q over \mathcal{I}* , denoted $\text{ans}(q, \mathcal{I})$, is the set of all n -tuples $\langle d_1, \dots, d_n \rangle \in \mathbf{C}^n$ such that $\langle d_1^{\mathcal{I}}, \dots, d_n^{\mathcal{I}} \rangle = \langle \pi(x_1), \dots, \pi(x_n) \rangle$ for some match π for q in \mathcal{I} . A *union of conjunctive queries* (UCQ) is a set of CQs with the same answer variable tuple. For a UCQ q , we let $\text{ans}(q, \mathcal{I}) = \bigcup_{q' \in q} \text{ans}(q', \mathcal{I})$. The *certain answer* to a CQ or a UCQ q over \mathcal{O} is defined as $\text{cert}(q, \mathcal{O}) = \{c \in \mathbf{C}^n \mid c \in \text{ans}(q, \mathcal{I}) \text{ for each model } \mathcal{I} \text{ of } \mathcal{O}\}$.

3 Explaining Negative Query Answers

We now define the problem considered in this paper:

Definition 1. Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an ontology, q a UCQ, and c a tuple of constants. We call $\mathcal{P} = \langle \mathcal{O}, q, c \rangle$ a query abduction problem (QAP). A solution to \mathcal{P} (or an explanation for \mathcal{P}) is any ABox \mathcal{U} such that the ontology $\mathcal{O}' = \langle \mathcal{T}, \mathcal{A} \cup \mathcal{U} \rangle$ is consistent and $c \in \text{cert}(q, \mathcal{O}')$. The set of all explanations for \mathcal{P} is denoted $\text{expl}(\mathcal{P})$.

If $c \notin \text{cert}(q, \mathcal{O})$, then we call c a *negative answer* to q over \mathcal{O} . Note that a query over the ontology can have a negative answer only if the ontology is satisfiable. Differently, if the ontology is unsatisfiable then the QAP \mathcal{P} does not have any solution. In the following, we will examine various restrictions to $\text{expl}(\mathcal{P})$ to reduce redundancy in explanations. This is achieved by the introduction of a preference relation among explanations. This relation is reflexive and transitive, i.e., we have a pre-order among solutions.

Definition 2. Assume a QAP \mathcal{P} . Let \preceq denote a pre-order on the set $\text{expl}(\mathcal{P})$ of solutions. We write $\mathcal{U} \prec \mathcal{U}'$ if $\mathcal{U} \preceq \mathcal{U}'$ and $\mathcal{U}' \not\preceq \mathcal{U}$. The preferred explanations $\text{expl}_{\preceq}(\mathcal{P})$ of a QAP \mathcal{P} under the pre-order \preceq are defined as follows: $\text{expl}_{\preceq}(\mathcal{P}) = \{\mathcal{U} \in \text{expl}(\mathcal{P}) \mid \text{there is no } \mathcal{U}' \in \text{expl}(\mathcal{P}) \text{ s.t. } \mathcal{U}' \prec \mathcal{U}\}$, i.e., $\text{expl}_{\preceq}(\mathcal{P})$ contains all the \preceq -explanations that are minimal under \preceq .

Two preference orders are considered here: the *subset-minimality order*, denoted by \subseteq , and the *minimum explanation size order*, denoted by \leq . The latter order is defined by $\mathcal{U} \leq \mathcal{U}'$ iff $|\mathcal{U}| \leq |\mathcal{U}'|$. Observe that $\text{expl}_{\leq}(\mathcal{P}) \subseteq \text{expl}_{\subseteq}(\mathcal{P})$.

We define now four decision problems related to (minimal) explanations, which are parametric w.r.t. the chosen preference order \preceq . Given a QAP \mathcal{P} :

- \preceq -EXISTENCE: Does there exist a \preceq -explanation for \mathcal{P} ?
- \preceq -RECOGNITION: Is a set \mathcal{U} of ABox assertions a \preceq -explanation for \mathcal{P} ?
- \preceq -RELEVANCE: Does an assertion α occur in some \preceq -explanation for \mathcal{P} ?
- \preceq -NECESSITY: Does an assertion α occur in all \preceq -explanations for \mathcal{P} ?

In the following, whenever no preference is applied (i.e., when \preceq is the identity) we omit to write \preceq in front of the problem's names. We provide an example, in which we highlight the consequences of choosing among the various orderings.

Table 1: Summary of main complexity results (completeness)

\preceq	\preceq -EXISTENCE	\preceq -RECOGNITION	\preceq -RELEVANCE	\preceq -NECESSITY
none	PTime (4.1)	NP	PTime (4.3)	PTime (4.2)
\leq	PTime	DP	$P_{ }^{NP}$	$P_{ }^{NP}$ (4.2)
\subseteq	PTime	DP	Σ_2^P (4.3)	PTime (4.2)

Example 1. Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an ontology describing a university domain, where \mathcal{T} is

$$\begin{aligned}
PostGrad &\sqsubseteq Student, & Tutor &\sqsubseteq Professor, & Advanced &\sqsubseteq Course, \\
UnderGrad &\sqsubseteq Student, & \exists hasTutor &\sqsubseteq PartTime, & \exists teaches &\sqsubseteq Professor, \\
UnderGrad &\sqsubseteq \neg PostGrad, & \exists hasTutor^- &\sqsubseteq Tutor, & \exists teaches^- &\sqsubseteq Course, \\
PartTime &\sqsubseteq UnderGrad.
\end{aligned}$$

That is, there are two different kinds of students, *PostGrad* and *UnderGrad*. Moreover, *PartTime* students are tutored by *Tutors*, who are particular professors. Additionally, the university offers some *Advanced* courses. Let the ABox \mathcal{A} consist of the assertions $teaches(rob, SWT)$, $hasTutor(peter, rob)$. Now, assume that a user is interested in finding all those who both teach an advanced course and tutor a student. Then, she would write the query

$$q(x) \leftarrow teaches(x, y), Advanced(y), hasTutor(z, x).$$

Moreover, she may expect *rob* to be part of the result, i.e., $rob \in cert(q, \mathcal{O})$, but this is not the case. Intuitively, *rob* satisfies all the constraints imposed by the query, except that the *SWT* course is not known to be *Advanced*. One can easily see that $\{teaches(rob, TOC), Advanced(TOC), hasTutor(john, rob)\}$ is an explanation, $\{teaches(rob, ALG), Advanced(ALG)\}$ is a \subseteq -minimal explanation, while $\{Advanced(SWT)\}$ is a \leq -minimal explanation. ■

In the next section, the complexity of the four main problems is studied in the light of the different preference relations.

4 Complexity of Explanations

Table 1 provides an overview of our complexity results. Recall that the class Σ_2^P is a member of the Polynomial Hierarchy [13]; it is the class of all decision problems solvable in non-deterministic polynomial time using an NP oracle. Moreover, the class $P_{||}^{NP}$ contains all the decision problems that can be solved in polynomial time with an NP oracle, where all oracle calls must be first prepared and then issued in parallel. The class DP contains all problems that, considered as languages, can be characterized as the intersection of a language in NP and a language in CONP [13]. Note that: $PTime \subseteq NP \subseteq DP \subseteq P_{||}^{NP} \subseteq \Sigma_2^P$ is believed to be a strict hierarchy of inclusions and here we make such an assumption.

Our results can be explained as follows. We show in the next section that EXISTENCE can be reduced to the PTIME-complete satisfiability problem for *DL-Lite_A* without the UNA [1], which justifies our PTIME upper bound. This result can then be used to characterize the complexity of RELEVANCE, NECESSITY, and \subseteq -NECESSITY. \leq -RELEVANCE and \leq -NECESSITY are harder. The reason being that in order to solve these problems one has to compute first the minimal size of a solution and, then, inspect all the solutions of that size. Additionally, there is another increase in complexity when dealing with \subseteq -RELEVANCE. The intuition is that there is an exponential number of candidate solutions to examine and for each of them one has to check that none of its subsets is itself a solution, which requires a CONP computation. Due to space limitations, the results on \preceq -RECOGNITION are not detailed in this paper (see [8] for more details). The intuition for the NP bound for RECOGNITION is that one needs simply to check consistency and perform query evaluation to solve the problem. In case a preference order is in place, one has to check minimality as well, which is a CONP check for \subseteq - and \leq -minimal explanations that leads to completeness for DP.

4.1 Complexity of \preceq -EXISTENCE

For \preceq -EXISTENCE note first that the existence of an explanation for \mathcal{P} implies the existence of an explanation under the \subseteq and \leq orderings. Thus, we only consider EXISTENCE. Our first task is to show that we can restrict ourselves to explanations built from the original signature of the input QAP plus a small number of fresh constants.

Proposition 1. *If $\mathcal{P} = \langle \mathcal{O}, q, c \rangle$ has a solution, then \mathcal{P} has a solution \mathcal{U}' with concepts, roles, and attributes only from \mathcal{O} and at most $m = \max_{q_i \in q} |at(q_i)|$ fresh ABox individuals.*

Proof. Assume an arbitrary solution \mathcal{U} to \mathcal{P} . Given the consistency of $\mathcal{O}' = \langle \mathcal{T}, \mathcal{A} \cup \mathcal{U} \rangle$, it follows that there exists a model \mathcal{I} of \mathcal{O}' under the UNA. W.l.o.g. we assume that $\Delta^{\mathcal{I}} = \mathbf{C}$ with $c^{\mathcal{I}} = c$ for each $c \in \mathbf{C}$. Additionally, the interpretation \mathcal{I} admits a match π for some $q_i(\mathbf{x}) \in q$, such that $\pi(\mathbf{x}) = c$. Let $\mathcal{U}' = \{A(o) \mid A(t) \in at(q_i) \text{ and } \pi(t) = o\} \cup \{R(o, o') \mid R(t, t') \in at(q_i) \text{ and } \pi(t) = o \text{ and } \pi(t') = o'\}$. Observe that \mathcal{U}' has no more individuals than q_i has variables. It remains to see that \mathcal{U}' is a solution. Clearly the original match π witnesses also $c \in \text{ans}(q_i, \mathcal{A} \cup \mathcal{U}')$. It remains to see that $\mathcal{O}'' = \langle \mathcal{T}, \mathcal{A} \cup \mathcal{U}' \rangle$ is consistent. But this follows from the fact that \mathcal{I} is a model of \mathcal{O}' and that the atoms in \mathcal{U}' hold in \mathcal{I} \square

The above restriction allows us to consider *canonical explanations*, i.e., explanations resulting from suitable instantiations of the bodies of CQs $q_i \in q$. Keeping in mind that CQs, seen as FOL formulae, are always satisfiable, an explanation does not exist only if the structure of the query is not compliant with the constraints expressed in the ontology. That is, for all the interpretations \mathcal{J} of q with $\text{ans}(q, \mathcal{J}) \neq \emptyset$, there is no model \mathcal{I} of \mathcal{O} , such that $\mathcal{I} \cup \mathcal{J} \models \mathcal{O}$. To check whether a UCQ is compliant with the ontological constraints, a naïve method is to iteratively go through all the CQs in q and instantiate them in the ABox. If for none of the CQs we obtain a consistent ontology, then the query violates some of the constraints imposed at the conceptual level.

Proposition 2. *For $DL\text{-Lite}_{\mathcal{A}}$ ontologies, EXISTENCE is PTIME-complete.*

Proof. (MEMBERSHIP) Note that $\mathcal{P} = \langle \mathcal{O}, q, \mathbf{c} \rangle$, with q a UCQ, has a solution iff $\mathcal{P}_{q'} = \langle \mathcal{O}, q', \mathbf{c} \rangle$ has a solution for some $q' \in q$. Hence, it suffices to show the upper bound for CQs. To this end, we provide a logspace reduction from EXISTENCE to consistency in $DL\text{-Lite}_{\mathcal{A}}$ without UNA, which in turn is PTIME-complete [1]. Assume a QAP $\mathcal{P} = \langle \mathcal{O}, q, \mathbf{c} \rangle$, where q is a CQ and $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$. We argue that \mathcal{P} has a solution iff $\mathcal{O}' = \langle \mathcal{T} \cup \mathcal{T}', \mathcal{A} \cup \mathcal{U}_q \cup \mathcal{A}' \rangle$ is consistent, where \mathcal{O}' is an ontology obtained from \mathcal{O} and $q(\mathbf{c})$ as follows. The ABox \mathcal{U}_q is obtained from $at(q(\mathbf{c}))$ by replacing each variable x with a fresh individual name a_x . The ABox \mathcal{A}' consists of assertions $A_o(o)$ for all constants o occurring in \mathcal{T}, \mathcal{A} and \mathcal{U}_q , where each A_o is a fresh concept name. The TBox \mathcal{T}' consists of axioms $A_o \sqsubseteq \neg A_{o'}$ for all pairs $o \neq o'$ of constants occurring in \mathcal{A} and $q(\mathbf{c})$. We now show that \mathcal{P} has a solution iff \mathcal{O}' is consistent.

Assume that \mathcal{P} has a solution \mathcal{U} . Then, due to consistency of $\mathcal{O}'' = \langle \mathcal{T}, \mathcal{A} \cup \mathcal{U} \rangle$, there is a model \mathcal{I} of \mathcal{O}'' under the UNA. Additionally, \mathcal{I} admits a match π for $q(\mathbf{c})$. Let \mathcal{I}' be the extension of \mathcal{I} that additionally interprets (i) constants in \mathcal{U}_q as $a_x^{\mathcal{I}'} = \pi(x)$ for all variables x in q , and (ii) $A_o^{\mathcal{I}'} = \{o^{\mathcal{I}'}\}$ for all freshly introduced A_o . It remains to show that \mathcal{I}' is a model of \mathcal{O}' . Observe that since \mathcal{I} is under the UNA, we have that $A_o^{\mathcal{I}'} \cap A_{o'}^{\mathcal{I}'} = \emptyset$ for all constant pairs $o \neq o'$. Thus \mathcal{I}' satisfies all the disjointness axioms $A_o \sqsubseteq \neg A_{o'}$ in \mathcal{T}' . The assertions in \mathcal{A}' are satisfied due to (ii), while the assertions in \mathcal{U}_q due to (i) above.

The other direction of the proof is obvious and we omit it here.

(HARDNESS) Let us reduce consistency in $DL\text{-Lite}_{\mathcal{A}}$ without UNA to EXISTENCE. Given $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, we create QAP $\mathcal{P} = \langle \mathcal{O}', q(), \langle \rangle \rangle$ simply by encoding the ABox \mathcal{A} in the CQ q by replacing each constant $a \in \mathcal{A}$ by a distinct variable name x_a in q , while the ontology \mathcal{O}' consists only of the TBox \mathcal{T} . \square

4.2 Complexity of (\sqsubseteq -)NECESSITY

The existence of an explanation is most of the times not very informative to the user. In fact, given a negative answer to a query, it is important to delineate the fundamental reasons leading to the absence of the expected tuple. That is, users would like to know which assertions occur in all the solutions to a QAP \mathcal{P} .

Proposition 3. *For $DL\text{-Lite}_{\mathcal{A}}$ ontologies, the NECESSITY and \sqsubseteq -NECESSITY problems are PTIME-complete.*

Proof. (MEMBERSHIP) We assume a QAP $\mathcal{P} = \langle \mathcal{O}, q, \mathbf{c} \rangle$ with $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, an assertion $\varphi(\mathbf{t})$ and consider NECESSITY first. This problem can be reduced to non-EXISTENCE, which was shown to be in PTIME in the previous section. We build $\mathcal{O}' = \langle \mathcal{T}', \mathcal{A}' \rangle$ such that $\varphi(\mathbf{t})$ occurs in all explanations for \mathcal{P} iff $\mathcal{P}' = \langle \mathcal{O}', q, \mathbf{c} \rangle$ has no explanation. We define \mathcal{O}' by setting $\mathcal{A}' = \mathcal{A} \cup \{\bar{\varphi}(\mathbf{t})\}$ and $\mathcal{T}' = \mathcal{T} \cup \{\bar{\varphi} \sqsubseteq \neg \varphi\}$, where $\bar{\varphi}$ is a fresh predicate name. It is easy to see that if \mathcal{P}' has no explanation, then either \mathcal{P} has no explanation as well, or, all the explanations for \mathcal{P} must contain $\varphi(\mathbf{t})$. For \sqsubseteq -NECESSITY, observe that $\varphi(\mathbf{t})$ occurs in all explanation for \mathcal{P} iff $\varphi(\mathbf{t})$ occurs in all \sqsubseteq -minimal explanations for \mathcal{P} . Thus \sqsubseteq -NECESSITY can be decided in polynomial time using our algorithm for NECESSITY.

(HARDNESS) The lower-bound can be proved through a logspace reduction from EXISTENCE to non-NECESSITY, that is, deciding whether there exists a solution to a QAP that does not contain the given assertion. Let $\mathcal{P} = \langle \mathcal{O}, q, c \rangle$ be a QAP with q a CQ, we build $\langle \mathcal{P}, \alpha \rangle$ such that \mathcal{P} has a solution iff $\langle \mathcal{P}, \alpha \rangle$ is a negative instance to NECESSITY. Let $\alpha = A(o)$, for some fresh concept A and constant o not occurring in \mathcal{P} . Now, assume \mathcal{P} has a solution \mathcal{U} . By Proposition 1, we know that there exists a solution \mathcal{U}' to \mathcal{P} containing only concept and role names from \mathcal{O} . Hence, $A(o) \notin \mathcal{U}'$, since concept name A is not in the ontology. Therefore, one can conclude that $A(o)$ is not a necessary assertion to \mathcal{P} .

The other direction is straightforward. □

Let us now show the complexity of necessity under the \leq preference order.

Theorem 1. *For $DL-Lite_{\mathcal{A}}$ ontologies, \leq -NECESSITY is $P_{\parallel}^{\text{NP}}$ -complete.*

Proof. (MEMBERSHIP only, HARDNESS in [8]) Let's assume a QAP $\mathcal{P} = \langle \mathcal{O}, q, c \rangle$ and an assertion α . By the use of canonical explanations, we know that the size m of the largest solution to \mathcal{P} corresponds to the size of the largest CQ in q . Observe that $\langle \mathcal{P}, \alpha \rangle$ is a negative instance of \leq -NECESSITY iff there is an $0 \leq i \leq m$ such that (a) \mathcal{P} has an explanation \mathcal{U} with $|\mathcal{U}| = i$ and $\alpha \notin \mathcal{U}$, and (b) \mathcal{U} is \leq -minimal. Thus, we use an auxiliary problem SIZE-OUT, which is to decide given a tuple $\langle \mathcal{P}', \alpha', n' \rangle$, where \mathcal{P}' is a QAP, α' is an assertion, and n' is an integer, whether there exists an explanation \mathcal{U}' for \mathcal{P}' such that $|\mathcal{U}'| = n'$ and $\alpha' \notin \mathcal{U}'$. Furthermore, the problem NO-SMALLER is to decide given a tuple $\langle \mathcal{P}', n' \rangle$ of a QAP and an integer whether there is no explanation \mathcal{U}' for \mathcal{P}' such that $|\mathcal{U}'| < n'$. Observe that SIZE-OUT is in NP, while NO-SMALLER is in CONP. Take the tuple $S = \langle A_0, B_0, \dots, A_m, B_m \rangle$, where (a) $A_i = \langle \mathcal{P}, \alpha, i \rangle$, for all $0 \leq i \leq m$, and (b) $B_i = \langle \mathcal{P}, i \rangle$, for all $0 \leq i \leq m$. Due to the above observation, α occurs in all \leq -minimal explanations \mathcal{U} for \mathcal{P} iff for all $0 \leq i \leq m$, one of the following holds: (i) A_i is a negative instance of SIZE-OUT, or (ii) B_i is a negative instance of NO-SMALLER. Note that S can be built in polynomial time in the size of the input, while the positivity of the instances in S can be decided by making $2m$ parallel calls to an NP oracle. Thus we obtain membership in $P_{\parallel}^{\text{NP}}$. □

4.3 Complexity of \preceq -RELEVANCE

A domain user faced with a negative answer to a query may ask herself whether, the absence of a certain ABox assertion α in the ontology is related with the lack of the tuple in the results. That is, she would like to know whether α occurs in some explanation to QAP \mathcal{P} .

Proposition 4. *For $DL-Lite_{\mathcal{A}}$ ontologies, RELEVANCE is PTIME-complete.*

Proof. (MEMBERSHIP) We assume a QAP $\mathcal{P} = \langle \mathcal{O}, q, c \rangle$ with $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and an assertion $\phi(\mathbf{t})$. We now provide a reduction from RELEVANCE to EXISTENCE. We construct $\mathcal{O}' = \langle \mathcal{T}, \mathcal{A}' \rangle$, where $\mathcal{A}' = \mathcal{A} \cup \phi(\mathbf{t})$. Then, \mathcal{P} has an explanation \mathcal{U} with $\phi(\mathbf{t}) \in \mathcal{U}$ iff there exists an explanation to $\mathcal{P}' = \langle \mathcal{O}', q, c \rangle$. This is because, any explanation to \mathcal{P}' can be extended by adding $\phi(\mathbf{t})$. It is simple to see that any such explanation is an

explanation to \mathcal{P} , as well. Finally, if \mathcal{P}' does not admit explanations, then $\phi(\mathbf{t})$ is a source of inconsistency in \mathcal{P} .

(HARDNESS) Hardness can again be proved with a reduction from EXISTENCE in a similar way as done in Section 4.2. \square

Let us now tackle the problem of \subseteq -RELEVANCE, for which we recall the FOL-rewritability of query answering in *DL-Lite_A*. Given an ABox \mathcal{A} , let $DB(\mathcal{A})$ be the following interpretation: (i) $\Delta^{DB(\mathcal{A})}$ is the set of constants occurring in \mathcal{A} , (ii) $A^{DB(\mathcal{A})} = \{o \in \Delta^{DB(\mathcal{A})} \mid A(o) \in \mathcal{A}\}$, for each atomic concept A , and (iii) $P^{DB(\mathcal{A})} = \{\langle o, o' \rangle \in \Delta^{DB(\mathcal{A})} \mid P(o, o') \in \mathcal{A}\}$, for each atomic role P .

Proposition 5 ([7]). *Let $PerfectRef(q, \mathcal{T})$ be the perfect reformulation of q w.r.t. \mathcal{T} , which is a UCQ obtained by applying the rewrite rules given in [7]. Then, $\text{cert}(q, \mathcal{O}) = \bigcup_{r \in PerfectRef(q, \mathcal{T})} \text{ans}(r, DB(\mathcal{A}))$.*

In other words, the certain answers to a UCQ can be computed by rewriting the query into a FOL query to be evaluated over the ABox.

Theorem 2. *For *DL-Lite_A* ontologies, \subseteq -RELEVANCE is Σ_2^P -complete.*

Proof. (MEMBERSHIP) The membership in Σ_2^P is clear from Algorithm 1, which works as follows. An explanation \mathcal{U} containing $\phi(\mathbf{t})$ is non-deterministically computed by guessing an instantiation of a subquery in $PerfectRef(q(\mathbf{c}), \mathcal{T})$, where $Anon$ is a set of fresh ABox individuals (see Proposition 1). Let HAS-SUBEXPL solve the problem of deciding whether a solution \mathcal{U} has a subset which is itself an explanation. The problem can be easily proved to be in NP. Then, the algorithm checks the complement of HAS-SUBEXPL in order to assure that none of the subsets of \mathcal{U} is itself an explanation, from which it follows that $\phi(\mathbf{t})$ is \subseteq -relevant. Checking the complement of HAS-SUBEXPL requires the power of a CONP machine. For this reason, the algorithm is solvable in non-deterministic polynomial time by a TM with an NP oracle.

(HARDNESS) We prove it by a reduction from the Σ_2^P -complete problem co-CERT3COL [15] (see also [4]). An instance of co-CERT3COL is given by a graph $G = (V, E)$ with vertices $V = \{0, \dots, n-1\}$ such that every edge is labeled with a disjunction of two literals over the Boolean variables $\{p_{(i,j)} \mid i, j < n\}$. G is a positive instance if there is a truth value assignment t to the Boolean variables such that the graph $t(G)$

Algorithm 1

INPUT: QAP $\mathcal{P} = \langle q, \mathcal{O}, \mathbf{c} \rangle$ and ABox assertion $\phi(\mathbf{t})$

OUTPUT: yes iff $\phi(\mathbf{t})$ is relevant to \mathcal{P}

- 1: Guess $q_i \in \{q_1, \dots, q_n\} = q$
 - 2: Guess the derivation of one rewriting $r(\mathbf{c})$ in $PerfectRef(q_i(\mathbf{c}), \mathcal{T})$
 - 3: Guess a set of atoms $U \subseteq at(r)$
 - 4: Guess a mapping π from $\mathbf{V}(q)$ to constants in $DB(\mathcal{A})$ and $Anon$
 - 5: Check that $(\mathcal{T}, \mathcal{A} \cup \mathcal{U}) \not\models \perp$, where \mathcal{U} is the instantiation of U through π .
 - 6: Check that $\phi(\mathbf{t}) \in \mathcal{U}$ and π is a match for $r(\mathbf{c})$ over $DB(\mathcal{A} \cup \mathcal{U})$
 - 7: Check that $\text{HAS-SUBEXPL}(\mathcal{P}, \mathcal{U}) = no$.
-

obtained from G by including only those edges whose label evaluates to true under t is not 3-colorable. Assume an instance G of co-CERT3COL. We show how to build in polynomial time a QAP $\mathcal{P}_G = \langle \langle \mathcal{T}_G, \mathcal{A}_G \rangle, q_G, \mathbf{c}_G \rangle$ and an ABox assertion α_G such that: G is a positive instance of co-CERT3COL iff α_G is \subseteq -relevant for \mathcal{P}_G . We use an empty TBox and a Boolean query, thus $\mathcal{T}_G = \emptyset$ and $\mathbf{c}_G = \langle \rangle$. The query q_G is a UCQ $q_G = q_{e_1} \cup \dots \cup q_{e_k} \cup q'$, where $\{e_1, \dots, e_k\} = E$, each q_{e_i} is an atomic query $q_{e_i}(x) \leftarrow W_{e_i}(x, y)$, and q' is defined as follows. Assume $\mathcal{B} = \{t, f\}$ to be the set of truth values. The query q' has the following atoms for each edge $e = (i, j)$ in E : (a) $B(x_i)$, $R_e(x_i, y_e)$, $R_e(y_e, x_j)$, $B(x_j)$, and (b) $P(y_e, z_{p_i})$, $A_{p_i}(z_{p_i})$, $W_{p_i}(z_{p_i}, z'_{p_i})$, where $p_i \in \{p_1, p_2\}$ and p_1, p_2 are the first and the second proposition in the labeling of e , respectively. The query q' simply incorporates G together with the disjunctions on the edges. Observe that if two edges have the same proposition in their label, then this will be reflected in q' by some shared variables z_{p_i} .

To build \mathcal{A}_G we use individuals c_p and $c_{\neg p}$ for the truth value of proposition p . Intuitively, the truth value of p will be determined by either $A_p(c_p)$ or $A_p(c_{\neg p})$ being in the update. Assume a tuple $\mathbf{t} = \langle e, v_1, v_2, a, b \rangle$, where $e \in E$, $\{v_1, v_2\} \subseteq \mathcal{B}$, and a, b are individuals. Let p_1, p_2 be the first and the second propositions of e . For $i \in \{1, 2\}$ and $v_i = \mathbf{t}$, let $l_i = p_i$ if p_i is positive and $l_i = \neg p_i$ otherwise. Similarly, for $i \in \{1, 2\}$ and $v_i = \mathbf{f}$, let $l_i = \neg p_i$ if p_i is positive and $l_i = p_i$ otherwise. Then, the ABox $\mathcal{A}(\mathbf{t})$ consists of the assertions $R_e(a, d_T)$, $R_e(d_T, b)$, $P(d_T, c_{l_1})$ and $P(d_T, c_{l_2})$ depending on the boolean values in input.

The ABox \mathcal{A}_G is the union of the following ABoxes:

- (A1) $\mathcal{A}(\langle e, v, v', a_i, a_j \rangle)$ for all $e \in E$, $v, v' \in \mathcal{B}$, $0 \leq i, j \leq 2$, and $i \neq j$;
- (A2) $\mathcal{A}(\langle e, f, f, a_i, a_i \rangle)$ for all $e \in E$, and $0 \leq i \leq 2$;
- (A3) $\mathcal{A}(\langle e, v, v', b, b \rangle)$ for all $e \in E$, $v, v' \in \mathcal{B}$;
- (A4) The ABox $\{B(a_0), B(a_1), B(a_2)\}$;
- (A5) The assertions $W_p(c_p, c_{\neg p})$ and $W_p(c_{\neg p}, c_p)$ for all propositions.

Let $\alpha_G = B(b)$. It is not too difficult to see that G is a positive instance of co-CERT3COL iff there exists an \subseteq -explanation \mathcal{U} to \mathcal{P} such that $\alpha_G \in \mathcal{U}$. Basically, definitions (A1)-(A3) encode a triangular structure T in which edges in G that evaluate to false according to a given truth assignment can be mapped on any edge of T , reflexive edges included. If an edge of G evaluates to true, then it must be mapped to one of the non-reflexive edges. This ensures that if G can be mapped to T under truth assignment t , then $t(G)$ is 3-colorable. Instead, definitions (A4)-(A5) define a cyclic structure C into which any graph G can be embedded. It has to be noted that the node b is not asserted to be a member of B , hence q_G cannot be mapped there directly with any truth assignment. We see this more formally next:

“ \Rightarrow ” Suppose there is a truth assignment t such that $t(G)$ is not 3-colorable. Let $\mathcal{U} = \{B(b)\} \cup \mathcal{U}_1$, where $\mathcal{U}_1 = \{A_p(c_p) \mid t(p) = t\} \cup \{A_p(c_{\neg p}) \mid t(p) = f\}$. It remains to argue that \mathcal{U} is a \subseteq -explanation to \mathcal{P} . It is not hard to see that \mathcal{U} is an explanation. Indeed q_G matches already in the ABox obtained by point (A3) (hint: since $B(b) \in \mathcal{U}$, we match q_G by mapping all variables of q_G to (interpretation of) b). Suppose there is a smaller update $\mathcal{U}' \subset \mathcal{U}$. Observe that $\mathcal{U}_1 \subseteq \mathcal{U}'$. This is because for all propositions p , the symbol A_p does not occur in \mathcal{A}_G but does occur in q_G . Then, $\mathcal{U} \setminus \{B(b)\}$ must be an

update. If this is the case, then q_G can be matched in \mathcal{A}_G without the ABox from (A3), i.e. in the triangle part. Then $t(G)$ is 3-colorable, which contradicts the assumption.

“ \Leftarrow ” Let \mathcal{U} be a \subseteq -minimal explanation containing $B(b)$. Due to the presence of $q_{e_1} \cup \dots \cup q_{e_k}$ in q_G and the assumption, the role W_p cannot occur in \mathcal{U} for any proposition p . Since \mathcal{U} is an explanation, by the definition of q' and (A5) we have that $A_p(c_p) \in \mathcal{U}$ or $A_p(c_{\neg p}) \in \mathcal{U}$ for all propositions p . Since for any proposition p we have that A_p occurs in q_G with one and only variable z_p , we know that exactly one of $A_p(c_p) \in \mathcal{U}$ and $A_p(c_{\neg p}) \in \mathcal{U}$ holds. Due to the atoms $W_p(z_p, z'_p)$ in q_G , we also have that individuals of the form c_p and $c_{\neg p}$ are the only ones that can get an A_p label. Consider the assignment t defined as follows: $t(e) = \text{t}$ if $A_p(c_p) \in \mathcal{U}$, and $t(e) = \text{f}$ if $A_p(c_{\neg p}) \in \mathcal{U}$. It is not difficult to argue that $t(G)$ is not 3-colorable and thus G is a positive instance of co-CERT3COL. Indeed, if $t(G)$ was 3-colorable, Q should be mappable into the triangle part obtained in (A1)-(A3). Then $\mathcal{U} \setminus \{B(b)\}$ would be a smaller update, which would mean a contradiction. \square

Note that the above lower bound holds already for empty TBoxes.

Proposition 6. *For DL-Lite $_{\mathcal{A}}$ ontologies, \leq -RELEVANCE is $P_{\parallel}^{\text{NP}}$ -complete.*

Proof. (MEMBERSHIP only, HARDNESS in [8]) \leq -RELEVANCE can be tackled in a way similar to \leq -NECESSITY. In fact, the algorithm described in Theorem 1 can be modified in order to solve this problem. Let SIZE-IN solve the following problem: given a tuple $\langle \mathcal{P}, \alpha, n \rangle$, where \mathcal{P} is a QAP, α an assertion, and n an integer, decide whether there exists an explanation \mathcal{U} , with $|\mathcal{U}| = n$ and $\alpha \in \mathcal{U}$. Then, we change the positivity condition of the \leq -NECESSITY algorithm as follows: α occurs in some \leq -minimal explanations \mathcal{U} for \mathcal{P} iff for some i , $0 \leq i \leq m$, it holds that: (i) A_i is a positive instance of SIZE-IN, and (ii) B_i is a positive instance of NO-SMALLER. It is easy to see that SIZE-IN is solvable in NP, hence the whole problem is again in $P_{\parallel}^{\text{NP}}$. \square

5 Conclusions

In this paper we have provided the formalization of a new problem, namely the explanation of negative answers to user queries over ontologies. A tuple is said to be a negative answer, if the user expects it to be part of $\text{cert}(q, \mathcal{O})$ but the tuple is actually not. In our framework, an explanation consists of an ABox that when added to the ontology leads the negative answer to be returned in the results of the query. We define various problems that help us in characterizing the complexity of finding explanations, such as the existence of explanations and relevance/necessity of assertions. We further consider a minimality criterion to be applied over explanations, such as subset-minimal and minimum-size preference orders. Within this framework, we provide a characterization of the computational complexity of the various problems for the DL *DL-Lite $_{\mathcal{A}}$* .

Future work includes studying the application of this framework to other lightweight description logics, starting with the \mathcal{EL} -family. We would also like to investigate the problem in the case where the ontology signature and the explanation signature may be different. That is, the signature over which explanations can be constructed is restricted only to a subset of the ontology signature [2].

References

1. A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyashev. The *DL-Lite* family and relations. *J. of Artificial Intelligence Research*, 36:1–69, 2009.
2. F. Baader, M. Bienvenu, C. Lutz, and F. Wolter. Query and predicate emptiness in description logics. In *Proc. of the 12th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR 2010)*, 2010.
3. F. Baader and R. Peñaloza. Axiom pinpointing in general tableaux. *J. of Logic and Computation*, 20(1):5–34, 2010.
4. P. A. Bonatti, C. Lutz, and F. Wolter. The complexity of circumscription in description logics. *J. of Artificial Intelligence Research*, 35:717–773, 2009.
5. A. Borgida, D. Calvanese, and M. Rodríguez-Muro. Explanation in the *DL-Lite* family of description logics. In *Proc. of the 7th Int. Conf. on Ontologies, DataBases, and Applications of Semantics (ODBASE 2008)*, volume 5332 of *Lecture Notes in Computer Science*, pages 1440–1457. Springer, 2008.
6. A. Borgida, E. Franconi, and I. Horrocks. Explaining \mathcal{ALC} subsumption. In *Proc. of the 14th Eur. Conf. on Artificial Intelligence (ECAI 2000)*, 2000.
7. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodríguez-Muro, and R. Rosati. Ontologies and databases: The *DL-Lite* approach. In S. Tessaris and E. Franconi, editors, *Semantic Technologies for Informations Systems – 5th Int. Reasoning Web Summer School (RW 2009)*, volume 5689 of *Lecture Notes in Computer Science*, pages 255–356. Springer, 2009.
8. D. Calvanese, M. Ortiz, M. Šimkus, and G. Stefanoni. The complexity of conjunctive query abduction in *DL-Lite*. Technical Report INFSYS RR 1843-11-01, Institute of Information Systems, Vienna University of Technology, 2011. Available at: <http://www.kr.tuwien.ac.at/research/reports/>.
9. A. Chapman and H. V. Jagadish. Why not? In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pages 523–534, 2009.
10. T. Eiter and G. Gottlob. The complexity of logic-based abduction. *J. of the ACM*, 42(1):3–42, 1995.
11. S. Klarman, U. Endriss, and S. Schlobach. ABox abduction in the description logic \mathcal{ALC} . *J. of Automated Reasoning*, 46(1):43–80, 2011.
12. D. L. McGuinness and A. Borgida. Explaining subsumption in description logics. In *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence (IJCAI'95)*, pages 816–821, 1995.
13. C. H. Papadimitriou. *Computational Complexity*. Addison Wesley Publ. Co., 1994.
14. R. Penaloza and B. Sertkaya. Complexity of axiom pinpointing in the *DL-Lite* family. In *Proc. of the 23rd Int. Workshop on Description Logic (DL 2010)*, volume 573 of *CEUR Electronic Workshop Proceedings*, <http://ceur-ws.org/>, 2010.
15. I. A. Stewart. Complete problems involving boolean labelled structures and projection transactions. *J. of Logic and Computation*, 1(6):861–882, 1991.