

Explaining Inconsistency-Tolerant Query Answering over Description Logic Knowledge Bases

Meghyn Bienvenu
CNRS, Univ. Montpellier, Inria
Montpellier, France

Camille Bourgaux
Univ. Paris-Sud, CNRS
Orsay, France

François Goasdoué
Univ. Rennes 1, CNRS
Lannion, France

Abstract

Several inconsistency-tolerant semantics have been introduced for querying inconsistent description logic knowledge bases. This paper addresses the problem of explaining why a tuple is a (non-)answer to a query under such semantics. We define explanations for positive and negative answers under the brave, AR and IAR semantics. We then study the computational properties of explanations in the lightweight description logic DL-Lite \mathcal{R} . For each type of explanation, we analyze the data complexity of recognizing (preferred) explanations and deciding if a given assertion is relevant or necessary. We establish tight connections between intractable explanation problems and variants of propositional satisfiability (SAT), enabling us to generate explanations by exploiting solvers for Boolean satisfaction and optimization problems. Finally, we empirically study the efficiency of our explanation framework using the well-established LUBM benchmark.

1 Introduction

Description logic (DL) knowledge bases (KBs) consist of a TBox (ontology) that provides conceptual knowledge about the application domain and an ABox (dataset) that contains facts about particular entities (Baader et al. 2003). The problem of querying such KBs using database-style queries (in particular, conjunctive queries) has been a major focus of recent DL research. Since scalability is a key concern, much of the work has focused on lightweight DLs for which query answering can be performed in polynomial time w.r.t. the size of the ABox. The DL-Lite family of lightweight DLs (Calvanese et al. 2007) is especially popular due to the fact that query answering can be reduced, via query rewriting, to the problem of standard database query evaluation.

Since the TBox is usually developed by experts and subject to extensive debugging, it is often reasonable to assume that its contents are correct. By contrast, the ABox is typically substantially larger and subject to frequent modifications, making errors almost inevitable. As such errors may render the KB inconsistent, several inconsistency-tolerant semantics have been introduced in order to provide meaningful answers to queries posed over inconsistent KBs. Arguably the most well-known is the *AR semantics* (Lembo et al. 2010), inspired by work on consistent query answering

in databases (cf. (Bertossi 2011) for a survey). Query answering under AR semantics amounts to considering those answers (w.r.t. standard semantics) that can be obtained from every *repair*, the latter being defined as an inclusion-maximal subset of the ABox that is consistent with the TBox. The more cautious *IAR semantics* (Lembo et al. 2010) queries the intersection of the repairs and provides a lower bound on AR semantics. The *brave semantics* (Bienvenu and Rosati 2013), which considers those answers holding in at least one repair, provides a natural upper bound.

The complexity of inconsistency-tolerant query answering in the presence of ontologies is now well understood (see e.g. (Rosati 2011; Bienvenu 2012; Lukasiewicz, Martinez, and Simari 2013)), so attention has turned to the problem of implementing these alternative semantics. There are currently two systems for querying inconsistent DL-Lite KBs: the QuID system of (Rosati et al. 2012) implements the IAR semantics, using either query rewriting or ABox cleaning, and our CQAPri system (2014) implements the AR, IAR and brave semantics, using tractable methods to obtain the answers under IAR and brave semantics and calls to a SAT solver to identify the answers holding under AR semantics.

The need to equip reasoning systems with explanation services is widely acknowledged by the DL community (see Section 6 for discussion and references), and such facilities are all the more essential when using inconsistency-tolerant semantics, as recently argued in (Arioua et al. 2014). Indeed, the brave, AR, and IAR semantics allow one to classify query answers into three categories of increasing reliability, and a user may naturally wonder why a given tuple was assigned to, or excluded from, one of these categories. In this paper, we address this issue by proposing and exploring a framework for explaining query answers under these three semantics. Our contributions are as follows:

- We define explanations of positive and negative query answers under brave, AR and IAR semantics. Intuitively, such explanations pinpoint the portions of the ABox that, in combination with the TBox, suffice to obtain the considered query answer. We focus on ABox assertions since inconsistencies are assumed to stem from errors in the ABox, and because this already yields a non-trivial framework to study.
- We investigate the main search and decision problems related to explanations: generating an (arbitrary) explanation, generating a most preferred explanation according to some

natural ranking criteria, recognizing (most preferred) explanations, and checking whether an assertion is relevant / necessary (i.e. appears in some / all explanations). We study the data complexity of these problems for DL-Lite \mathcal{R} , showing (in)tractability of each of the tasks and pinpointing the exact complexity of the intractable decision problems.

- We establish tight connections between the intractable decision problems, as well as the problem of generating (preferred) explanations, and SAT-based reasoning tasks. This enables effective solutions to these problems using solvers for Boolean satisfaction and optimization problems.
- Finally, we present an implementation of our explanation services on top of the CQAPri system and SAT4J solver. Using the LUBM benchmark, we show its practical interest on a large ABox with increasing number of conflicting assertions: explanations of answers are computed rapidly overall, typically in a few milliseconds, rarely above a second.

2 Preliminaries

We briefly recall the syntax and semantics of DLs, and the inconsistency-tolerant semantics we use.

Syntax A DL *knowledge base* (KB) consists of an ABox and a TBox, both constructed from a set N_C of *concept names* (unary predicates), a set N_R of *role names* (binary predicates), and a set N_I of *individuals* (constants). The ABox (dataset) consists of a finite number of *concept assertions* of the form $A(a)$ and *role assertions* of the form $R(a, b)$, where $A \in N_C$, $R \in N_R$, $a, b \in N_I$. The TBox (ontology) consists of a set of axioms whose form depends on the DL in question. In DL-Lite \mathcal{R} , TBox axioms are either *concept inclusions* $B \sqsubseteq C$ or *role inclusions* $Q \sqsubseteq S$ formed according to the following syntax (where $A \in N_C$ and $R \in N_R$):

$$B := A \mid \exists Q, C := B \mid \neg B, Q := R \mid R^-, S := Q \mid \neg Q$$

Semantics An *interpretation* has the form $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set and $\cdot^{\mathcal{I}}$ maps each $a \in N_I$ to $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each $A \in N_C$ to $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, and each $R \in N_R$ to $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ is straightforwardly extended to general concepts and roles, e.g. $(R^-)^{\mathcal{I}} = \{(c, d) \mid (d, c) \in R^{\mathcal{I}}\}$ and $(\exists Q)^{\mathcal{I}} = \{c \mid \exists d : (c, d) \in Q^{\mathcal{I}}\}$. We say that \mathcal{I} satisfies an inclusion $G \sqsubseteq H$ if $G^{\mathcal{I}} \subseteq H^{\mathcal{I}}$; it satisfies $A(a)$ (resp. $R(a, b)$) if $a^{\mathcal{I}} \in A^{\mathcal{I}}$ (resp. $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$). We call \mathcal{I} a *model* of $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ if \mathcal{I} satisfies all axioms in \mathcal{T} and assertions in \mathcal{A} . A KB \mathcal{K} is *consistent* if it has a model; otherwise it is inconsistent, denoted $\mathcal{K} \models \perp$. An ABox \mathcal{A} is \mathcal{T} -consistent if the KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is consistent.

Example 1. As a running example, we consider a simple KB $\mathcal{K}_{\text{ex}} = (\mathcal{T}_{\text{ex}}, \mathcal{A}_{\text{ex}})$ about the university domain that contains concepts for postdoctoral researchers (Postdoc), professors (Pr) of two levels of seniority (APr, FPr), and PhD holders (PhD), as well as roles to link advisors to their students (Adv) and instructors to their courses (Teach). The ABox \mathcal{A}_{ex} provides information about an individual a :

$$\begin{aligned} \mathcal{T}_{\text{ex}} = \{ & \text{Postdoc} \sqsubseteq \text{PhD}, \text{Pr} \sqsubseteq \text{PhD}, \text{Postdoc} \sqsubseteq \neg \text{Pr}, \\ & \text{FPr} \sqsubseteq \text{Pr}, \text{APr} \sqsubseteq \text{Pr}, \text{APr} \sqsubseteq \neg \text{FPr}, \exists \text{Adv} \sqsubseteq \text{Pr} \} \\ \mathcal{A}_{\text{ex}} = \{ & \text{Postdoc}(a), \text{FPr}(a), \text{APr}(a), \text{Adv}(a, b), \end{aligned}$$

$$\text{Teach}(a, c_1), \text{Teach}(a, c_2), \text{Teach}(a, c_3) \}$$

Observe that \mathcal{A}_{ex} is \mathcal{T}_{ex} -inconsistent.

Queries We focus on *conjunctive queries* (CQs) which take the form $\exists \vec{y} \psi$, where ψ is a conjunction of atoms of the forms $A(t)$ or $R(t, t')$, t, t' are variables or individuals, and \vec{y} is a tuple of variables from ψ . When we use the generic term *query*, we mean a CQ. Given a CQ q with free variables x_1, \dots, x_k and a tuple of individuals $\vec{a} = (a_1, \dots, a_k)$, we use $q(\vec{a})$ to denote the first-order sentence resulting from replacing each x_i by a_i . A tuple \vec{a} is a *certain answer* to q over \mathcal{K} , written $\mathcal{K} \models q(\vec{a})$, iff $q(\vec{a})$ holds in every model of \mathcal{K} .

Causes and conflicts A *cause* for $q(\vec{a})$ w.r.t. KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is a minimal \mathcal{T} -consistent subset $\mathcal{C} \subseteq \mathcal{A}$ such that $\mathcal{T}, \mathcal{C} \models q(\vec{a})$. We use $\text{causes}(q(\vec{a}), \mathcal{K})$ to refer to the set of causes for $q(\vec{a})$. A *conflict* for \mathcal{K} is a minimal \mathcal{T} -inconsistent subset of \mathcal{A} , and $\text{confl}(\mathcal{K})$ denotes the set of conflicts for \mathcal{K} .

When \mathcal{K} is a DL-Lite \mathcal{R} KB, every conflict for \mathcal{K} has at most two assertions. We can thus define the set of *conflicts of a set of assertions* $\mathcal{C} \subseteq \mathcal{A}$ as follows:

$$\text{confl}(\mathcal{C}, \mathcal{K}) = \{\beta \mid \exists \alpha \in \mathcal{C}, \{\alpha, \beta\} \in \text{confl}(\mathcal{K})\}.$$

Inconsistency-tolerant semantics A *repair* of $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is an inclusion-maximal subset of \mathcal{A} that is \mathcal{T} -consistent. We consider three previously studied inconsistency-tolerant semantics based upon repairs. Under *AR semantics*, a tuple \vec{a} is an answer to q over \mathcal{K} , written $\mathcal{K} \models_{\text{AR}} q(\vec{a})$, just in the case that $\mathcal{T}, \mathcal{R} \models q(\vec{a})$ for *every* repair \mathcal{R} of \mathcal{K} (equivalently: every repair contains some cause of $q(\vec{a})$). If there exists *some* repair \mathcal{R} such that $\mathcal{T}, \mathcal{R} \models q(\vec{a})$ (equivalently: $\text{causes}(q(\vec{a}), \mathcal{K}) \neq \emptyset$), then \vec{a} is an answer to q under *brave semantics*, written $\mathcal{K} \models_{\text{brave}} q(\vec{a})$. For *IAR semantics*, we have $\mathcal{K} \models_{\text{IAR}} q(\vec{a})$ iff $\mathcal{T}, \mathcal{R}_{\cap} \models q(\vec{a})$ (equivalently, \mathcal{R}_{\cap} contains some cause for $q(\vec{a})$), where \mathcal{R}_{\cap} is the *intersection of all repairs* of \mathcal{K} . The three semantics are related as follows:

$$\mathcal{K} \models_{\text{IAR}} q(\vec{a}) \Rightarrow \mathcal{K} \models_{\text{AR}} q(\vec{a}) \Rightarrow \mathcal{K} \models_{\text{brave}} q(\vec{a})$$

For $S \in \{\text{AR}, \text{brave}, \text{IAR}\}$, we call \vec{a} a (*positive*) S -*answer* (resp. *negative S-answer*) if $\mathcal{K} \models_S q(\vec{a})$ (resp. $\mathcal{K} \not\models_S q(\vec{a})$).

Example 2. The example KB \mathcal{K}_{ex} has three repairs:

$$\begin{aligned} \mathcal{R}_1 &= \mathcal{A}_{\text{ex}} \setminus \{\text{FPr}(a), \text{APr}(a), \text{Adv}(a, b)\} \\ \mathcal{R}_2 &= \mathcal{A}_{\text{ex}} \setminus \{\text{Postdoc}(a), \text{FPr}(a)\} \\ \mathcal{R}_3 &= \mathcal{A}_{\text{ex}} \setminus \{\text{Postdoc}(a), \text{APr}(a)\} \end{aligned}$$

We consider the following example queries: $q_1 = \text{Prof}(x)$, $q_2 = \exists y \text{PhD}(x) \wedge \text{Teach}(x, y)$, and $q_3 = \exists y \text{Teach}(x, y)$. Evaluating these queries on \mathcal{K}_{ex} yields the following results:

$$\begin{array}{lll} \mathcal{K}_{\text{ex}} \models_{\text{brave}} q_1(a) & \mathcal{K}_{\text{ex}} \models_{\text{AR}} q_2(a) & \mathcal{K}_{\text{ex}} \models_{\text{IAR}} q_3(a) \\ \mathcal{K}_{\text{ex}} \not\models_{\text{AR}} q_1(a) & \mathcal{K}_{\text{ex}} \not\models_{\text{IAR}} q_2(a) & \end{array}$$

3 Explaining Query Results

The aforementioned inconsistency-tolerant semantics allows us to identify three types of positive query answer:

$$\text{IAR-answers} \subseteq \text{AR-answers} \subseteq \text{brave-answers}$$

The goal of the present work is to help the user understand the classification of a particular tuple, e.g. why is \vec{a}

an AR-answer, and why is it not an IAR-answer? To this end, we introduce the notion of *explanation* for positive and negative query answers under brave, AR, and IAR semantics. For consistent KBs, these three semantics collapse into the classical one, so existing explanation frameworks can be used (Borgida, Calvanese, and Rodriguez-Muro 2008; Calvanese et al. 2013; Du, Wang, and Shen 2014).

Formally, the explanations we consider will take either the form of a set of ABox assertions (viewed as a conjunction) or a set of sets of assertions (interpreted as a disjunction of conjunctions). We chose to focus on ABox assertions, rather than TBox axioms, since we target scenarios in which inconsistencies are due to errors in the ABox, so understanding the link between (possibly faulty) ABox assertions and query results is especially important. Moreover, as we shall see in Sections 4 and 5, our ‘ABox-centric’ explanation framework already poses non-trivial computational challenges.

The simplest answers to explain are positive brave- and IAR-answers. We can use the query’s causes as explanations for the former, and the causes that do not participate in any contradiction for the latter. Note that in what follows we suppose that $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is a KB and q is a query.

Definition 1. An explanation for $\mathcal{K} \models_{\text{brave}} q(\vec{a})$ is a cause for $q(\vec{a})$ w.r.t. \mathcal{K} . An explanation for $\mathcal{K} \models_{\text{IAR}} q(\vec{a})$ is a cause \mathcal{C} for $q(\vec{a})$ w.r.t. \mathcal{K} such that $\mathcal{C} \subseteq \mathcal{R}$ for every repair \mathcal{R} of \mathcal{K} .

Example 3. There are three explanations for $\mathcal{K}_{\text{ex}} \models_{\text{brave}} q_1(a)$: $\text{FPr}(a)$, $\text{APr}(a)$, and $\text{Adv}(a, b)$. There are twelve explanations for $\mathcal{K}_{\text{ex}} \models_{\text{brave}} q_2(a)$: $\text{Postdoc}(a) \wedge \text{Teach}(a, c_j)$, $\text{FPr}(a) \wedge \text{Teach}(a, c_j)$, $\text{APr}(a) \wedge \text{Teach}(a, c_j)$, and $\text{Adv}(a, b) \wedge \text{Teach}(a, c_j)$, for each $j \in \{1, 2, 3\}$. There are three explanations for $\mathcal{K}_{\text{ex}} \models_{\text{IAR}} q_3(a)$: $\text{Teach}(a, c_1)$, $\text{Teach}(a, c_2)$, and $\text{Teach}(a, c_3)$.

To explain why a tuple is an AR-answer, it is no longer sufficient to give a single cause, since different repairs may use different causes. We will therefore define explanations as (minimal) disjunctions of causes that ‘cover’ all repairs.

Definition 2. An explanation for $\mathcal{K} \models_{\text{AR}} q(\vec{a})$ is a set $\mathcal{E} = \{\mathcal{C}_1, \dots, \mathcal{C}_m\} \subseteq \text{causes}(q(\vec{a}), \mathcal{K})$ such that (i) every repair \mathcal{R} of \mathcal{K} contains some \mathcal{C}_i , and (ii) no proper subset of \mathcal{E} satisfies this property.

Example 4. There are 36 explanations for $\mathcal{K}_{\text{ex}} \models_{\text{AR}} q_2(a)$, each taking one of the following two forms:

$$\begin{aligned} \mathcal{E}_{ij} &= (\text{Postdoc}(a) \wedge \text{Teach}(a, c_i)) \vee (\text{Adv}(a, b) \wedge \text{Teach}(a, c_j)) \\ \mathcal{E}'_{ijk} &= (\text{Postdoc}(a) \wedge \text{Teach}(a, c_i)) \vee (\text{FPr}(a) \wedge \text{Teach}(a, c_j)) \\ &\quad \vee (\text{APr}(a) \wedge \text{Teach}(a, c_k)) \end{aligned}$$

for some $i, j, k \in \{1, 2, 3\}$.

We next consider how to explain negative AR- and IAR-answers, i.e., brave-answers not entailed under AR or IAR semantics. For AR semantics, the idea is to give a (minimal) subset of the ABox that is consistent with the TBox and contradicts every cause of the query, since any such subset can be extended to a repair that omits all causes. For IAR semantics, the formulation is slightly different as we need only ensure that every cause is contradicted by some consistent subset, as this shows that no cause belongs to all repairs.

Definition 3. An explanation for $\mathcal{K} \not\models_{\text{AR}} q(\vec{a})$ is a \mathcal{T} -consistent subset $\mathcal{E} \subseteq \mathcal{A}$ such that: (i) $\mathcal{T}, \mathcal{E} \cup \mathcal{C} \models \perp$ for every $\mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K})$, (ii) no proper subset of \mathcal{E} has this property. An explanation for $\mathcal{K} \not\models_{\text{IAR}} q(\vec{a})$ is a (possibly \mathcal{T} -inconsistent) subset $\mathcal{E} \subseteq \mathcal{A}$ such that: (i) for every $\mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K})$, there exists a \mathcal{T} -consistent subset $\mathcal{E}' \subseteq \mathcal{E}$ with $\mathcal{T}, \mathcal{E}' \cup \mathcal{C} \models \perp$, (ii) no $\mathcal{E}' \subsetneq \mathcal{E}$ has this property.

Example 5. The unique explanation for $\mathcal{K}_{\text{ex}} \not\models_{\text{AR}} q_1(a)$ is $\text{Postdoc}(a)$, which contradicts the three causes of $q_1(a)$. The explanations for $\mathcal{K}_{\text{ex}} \not\models_{\text{IAR}} q_2(a)$ are: $\text{FPr}(a) \wedge \text{Postdoc}(a)$, $\text{APr}(a) \wedge \text{Postdoc}(a)$, and $\text{Adv}(a, b) \wedge \text{Postdoc}(a)$, where the first assertion of each explanation contradicts the causes of $q_2(a)$ that contain $\text{Postdoc}(a)$, and the second one contradicts those that contain $\text{FPr}(a)$, $\text{APr}(a)$ or $\text{Adv}(a, b)$.

When there are a large number of explanations for a given answer, it may be impractical to present them all to the user. In such cases, one may choose to rank the explanations according to some preference criteria, and to present one or a small number of most *preferred explanations*. In this work, we will use *cardinality* to rank explanations for brave- and IAR-answers and negative AR- and IAR-answers. For positive AR-answers, we consider two ranking criteria: the *number of disjuncts*, and the *total number of assertions*.

Example 6. Reconsider explanations \mathcal{E}_{11} and \mathcal{E}'_{123} for $\mathcal{K}_{\text{ex}} \models_{\text{AR}} q_2(a)$. There are at least two reasons why \mathcal{E}_{11} may be considered easier to understand than \mathcal{E}'_{123} . First, \mathcal{E}_{11} contains fewer disjuncts, hence requires less disjunctive reasoning. Second, both disjuncts of \mathcal{E}_{11} use the same Teach assertion, whereas \mathcal{E}'_{123} uses three different Teach assertions, which may lead the user to (wrongly) believe all are needed to obtain the query result. Preferring explanations having the fewest number of disjuncts, and among them, those involving a minimal set of assertions, leads to focusing on the explanations of the form \mathcal{E}_{ii} , where $i \in \{1, 2, 3\}$.

A second complementary approach is to concisely summarize the set of explanations in terms of the *necessary assertions* (i.e. appearing in every explanation) and the *relevant assertions* (i.e. appearing in at least one explanation).

Example 7. If we tweak the example KB to include n courses taught by a , then there would be $n^2 + n^3$ explanations for $\mathcal{K}_{\text{ex}} \models_{\text{AR}} q_2(a)$, built using only $n + 4$ assertions. Presenting the necessary assertions (here: $\text{Postdoc}(a)$) and relevant ones ($\text{FPr}(a)$, $\text{APr}(a)$, $\text{Adv}(a, b)$, $\text{Teach}(a, c_i)$) gives a succinct overview of the set of explanations.

4 Complexity Analysis

We next study the computational properties of the different notions of explanation defined in Section 3. In addition to the problem of generating a single explanation (GENONE), or a single best explanation (GENBEST) according to a given criteria, we consider four related decision problems: decide whether a given assertion appears in some explanation (REL) or in every explanation (NEC), decide whether a candidate is an explanation (REC), resp. a best explanation (BEST REC).

In the remainder of the paper, we focus on KBs expressed in the lightweight logic DL-Lite $_{\mathcal{R}}$ since it is a popular choice for ontology-based data access and the only DL for which

| | brave, IAR | AR | neg. AR | neg. IAR |
|-----------------------|------------|------------------------------|----------|----------|
| GENONE | in P | NP-h | NP-h | in P |
| GENBEST [†] | in P | Σ_2^p -h [‡] | NP-h | NP-h* |
| REL | in P | Σ_2^p -co | NP-co | in P |
| NEC | in P | NP-co | coNP-co | in P |
| REC | in P | BH ₂ -co | in P | in P |
| BEST REC [†] | in P | Π_2^p -co [‡] | coNP-co* | coNP-co* |

[†] upper bounds hold for ranking criteria that can be decided in P

[‡] lower bounds hold for smallest disjunction or fewest assertions

* lower bounds hold for cardinality-minimal explanations

Figure 1: Data complexity results for CQs.

the three considered semantics have been implemented. As we target applications in which the ABox is significantly larger than the TBox and query, we use the *data complexity* measure, which is only with respect to the size of the ABox.

Our complexity results are displayed in Figure 1.

Theorem 1. *The results in Figure 1 hold.*

In what follows, we present some key ideas underlying Theorem 1 (detailed proofs are provided in the appendix).

Positive brave- and IAR-answers We recall that in DL-Lite_R, KB satisfiability and query answering are in P w.r.t. data complexity (Calvanese et al. 2007), and conflicts are of size at most two. It follows that the causes and their conflicts can be computed in P w.r.t. data complexity (using e.g. standard query rewriting algorithms). From the causes and conflicts, we can immediately read off the explanations for brave- and IAR-answers; a simple examination of the set of explanations enables us to solve the decision problems.

Positive AR-answers We relate explanations of AR-answers to minimal unsatisfiable subsets of a set of propositional clauses. Let us recall that, given sets F and H of soft and hard clauses respectively, a subset $M \subseteq F$ is a *minimal unsatisfiable subset* (MUS) of F w.r.t. H if (i) $M \cup H$ is unsatisfiable, and (ii) $M' \cup H$ is satisfiable for every $M' \subsetneq M$.

To explain $\mathcal{K} \models_{AR} q(\vec{a})$, we consider the soft clauses

$$\varphi_{-q} = \{\lambda_C \mid C \in \text{causes}(q(\vec{a}), \mathcal{K})\} \text{ with } \lambda_C = \bigvee_{\beta \in \text{confl}(C, \mathcal{K})} x_\beta$$

and the hard clauses

$$\varphi_{cons} = \{\neg x_\alpha \vee \neg x_\beta \mid x_\alpha, x_\beta \in \text{vars}(\varphi_{-q}), \{\alpha, \beta\} \in \text{confl}(\mathcal{K})\}$$

It was proven by Bienvenu et al. (2014) that $\mathcal{K} \models_{AR} q(\vec{a})$ iff $\varphi_{-q} \cup \varphi_{cons}$ is unsatisfiable, and we can further show:

Proposition 1. *A set $\mathcal{E} \subseteq \text{causes}(q(\vec{a}), \mathcal{K})$ is an explanation for $\mathcal{K} \models_{AR} q(\vec{a})$ iff $\{\lambda_C \mid C \in \mathcal{E}\}$ is a MUS of φ_{-q} w.r.t. φ_{cons} .*

In addition to enabling us to exploit MUS algorithms for explanation generation, Proposition 1 can be combined with known complexity results for MUSes (Liberatore 2005) to infer the upper bounds for REL, REC, and BEST REC. Moreover, we can find a TBox \mathcal{T}^* and assertion α^* such that for every unsatisfiable clause set φ , we can construct

in polytime an ABox A_φ such that the explanations for $(\mathcal{T}^*, A_\varphi) \models_{AR} \alpha^*$ correspond (in a precise sense) to the MUSes of φ (w.r.t. \emptyset). This reduction enables us to transfer complexity lower bounds for MUSes to our setting.

Negative AR-answers We relate explanations of negative AR-answers to minimal models of $\varphi_{-q} \cup \varphi_{cons}$. Given a clause set ψ over variables X , a set $M \subseteq X$ is a *minimal model* of ψ iff (i) every valuation that assigns true to all variables in M satisfies ψ , (ii) no $M' \subsetneq M$ satisfies this condition. Cardinality-minimal models are defined analogously.

Proposition 2. *A set \mathcal{E} is an explanation (resp. cardinality-minimal explanation) for $\mathcal{K} \not\models_{AR} q(\vec{a})$ iff $\{x_\alpha \mid \alpha \in \mathcal{E}\}$ is a minimal (resp. cardinality-minimal) model of $\varphi_{-q} \cup \varphi_{cons}$.*

The preceding result enables us to generate explanations using tools for computing (cardinality-)minimal models; it also yields a coNP upper bound for NEC since α belongs to all explanations just in the case that $\varphi_{-q} \cup \varphi_{cons} \cup \{\neg x_\alpha\}$ is unsatisfiable. Recognizing an explanation \mathcal{E} can be done in P by checking consistency of \mathcal{E} , inconsistency of $\mathcal{E} \cup \mathcal{C}$ for every cause \mathcal{C} , and minimality of \mathcal{E} . It follows that REL is in NP and BEST REC in coNP for ranking criteria that can be decided in P (guess an explanation that contains the assertion, resp. is a better explanation). The NP and coNP lower bounds are proved by reductions from (UN)SAT.

Negative IAR-answers We relate explanations of negative IAR-answers to minimal models of the clause set φ_{-q} .

Proposition 3. *A set \mathcal{E} is an explanation (resp. cardinality-minimal explanation) for $\mathcal{K} \not\models_{IAR} q(\vec{a})$ iff $\{x_\alpha \mid \alpha \in \mathcal{E}\}$ is a minimal (resp. cardinality-minimal) model of φ_{-q} .*

Importantly, φ_{-q} does not contain any negative literals, and it is known that for *positive* clause sets, a single minimal model can be computed in P, and the associated relevance problem is also in P. The intractability results for GENBEST and BEST REC are obtained by relating cardinality-minimal models of monotone 2CNF formulas to cardinality-minimal explanations of negative IAR-answers.

5 Prototype and experiments

We implemented our explanation framework in Java within our CQAPri system (www.lri.fr/~bourgaux/CQAPri), which supports querying of DL-Lite_R KBs under several inconsistency-tolerant semantics, including the brave, AR and IAR semantics. We used the SAT4J v2.3.4 SAT solver (www.sat4j.org) to compute MUSes and cardinality-minimal models (Berre and Parrain 2010).

CQAPri classifies a query answer \vec{a} into one of 3 classes:

- **Possible:** $\mathcal{K} \models_{brave} q(\vec{a})$ and $\mathcal{K} \not\models_{AR} q(\vec{a})$
- **Likely:** $\mathcal{K} \models_{AR} q(\vec{a})$ and $\mathcal{K} \not\models_{IAR} q(\vec{a})$
- **(Almost) sure:** $\mathcal{K} \models_{IAR} q(\vec{a})$

Explaining the answer \vec{a} consists in providing *all* the explanations for \vec{a} being a positive answer under the first semantics and a *single* explanation for it being a negative answer under the other one (i.e. a counter-example), together with the necessary and relevant assertions. Positive explanations

| Query id | shape | #atoms | #variables | #rewritings |
|----------|--------|--------|------------|-------------|
| g2 | atomic | 1 | 1 | 44 |
| g3 | atomic | 1 | 1 | 44 |
| q1 | dag | 5 | 2 | 6401 |
| q2 | tree | 3 | 2 | 450 |
| q3 | tree | 2 | 3 | 155 |
| q4 | dag | 6 | 4 | 202579 |

Table 1: Queries in term of shape, number of atoms, number of variables, and number of CQs in the UCQ-rewriting.

are ranked as explained in Section 3; for ranking positive-AR answers, the user can choose the priority between the numbers of disjuncts and the total number of assertions.

Explanations are computed using the results on positive and negative answers from Section 4. We thus need the *causes* of the query answers as well as their *conflicts*. The conflicts are directly available from CQAPri; for the causes, CQAPri uses query rewriting to identify consistent (but not necessarily minimal) subsets of the ABox entailing the answers, from which we must prune non-minimal ones.

To assess the practical interest of our framework, we empirically study the properties of our implementation, in particular: the impact of varying the percentage of assertions in conflict, the typical number and size of explanations, and the extra effort required to generate cardinality-minimal explanations for negative IAR-answers rather than arbitrary ones.

Experimental setting We used the CQAPri benchmark available at www.lri.fr/~bourgaux/CQAPri, which builds on the DL-Lite_R version (Lutz et al. 2013) of the Lehigh University Benchmark (swat.cse.lehigh.edu/projects/lubm). It extends the DL-Lite_R TBox with negative inclusions and describes how to obtain an ABox with a natural repartition of conflicts by adding assertions to an initial ABox consistent with the enriched TBox.

We used a consistent database with 100 universities (more than 10 million assertions) from which we generated seven inconsistent ABoxes with different ratios of assertions in conflicts by adding from 8005 to 351724 assertions. These ABoxes are denoted cX , with X the ratio of conflicts varying from 5% to a value of 44% challenging our approach. Also, the way we generate conflicts ensures $cX \subseteq cY$ if $X \leq Y$.

The queries can be found on the CQAPri website. Table 1 displays the characteristics of these queries, which have (i) a variety of structural aspects and number of rewritings, and (ii) answers in the three considered classes (see Table 2).

Our hardware is an Intel Xeon X5647 at 2.93 GHz with 16 GB of RAM, running CentOS 6.7. Reported times are averaged over 5 runs.

Experimental results We summarize below the general tendencies we observed. Table 2 shows the number of answers from each class for each query, as well as the distribution of the explanation times for these answers. Figure 2 shows the proportion of time spent in the different phases and the total time to explain *all* query answers over ABoxes of increasing difficulty. The explanation cost, given by the upper bar, consists in pruning non-minimal consistent subsets of the ABox entailing the answers to get the causes, and computing the

| | #answers | < 10 | [10, 100[| [100, 1000[| > 1000 |
|-----------|----------|--------|-----------|-------------|--------|
| g2 Sure | 29074 | 100 | 0 | 0 | 0 |
| g2 Likely | 1166 | 86.2 | 0 | 6.4 | 7.4 |
| g2 Poss. | 18162 | 52.2 | 2.7 | 39 | 6.1 |
| g3 Sure | 129083 | 99.999 | 0.001 | 0 | 0 |
| g3 Likely | 8902 | 99.98 | 0.02 | 0 | 0 |
| g3 Poss. | 19737 | 98.58 | 0.05 | 1.36 | 0.01 |
| q1 Likely | 10 | 50 | 50 | 0 | 0 |
| q2 Poss. | 133 | 100 | 0 | 0 | 0 |
| q3 Poss. | 208752 | 99.578 | 0.421 | 0 | 0.001 |
| q4 Sure | 128616 | 99.99 | 0.01 | 0 | 0 |
| q4 Likely | 192 | 99.5 | 0.5 | 0 | 0 |
| q4 Poss. | 64820 | 87.055 | 12.942 | 0 | 0.003 |

Table 2: Number of answers of each class and distribution of explanation times (in ms) per query on $c29$.

explanations from the causes and conflicts. The two lower bars relate to the query answering phase, which consists in rewriting and executing the query (execute), and identifying Sure, Likely, and Possible answers (classify).

The main conclusion is that explaining a *single* query answer, as described above, is always feasible and fast ($\leq 1s$) when there are a *few* percent of conflicts in the ABox ($c5$ case), as is likely to be the case in most real applications. Even with a *significant* percentage of conflicts ($c29$ case, Table 2), the longest time observed is below 8s. In *all* the experiments we made, explaining a *single* answer typically takes less than 10ms, rarely more than 1s. (Computing explanations of *all* answers can be prohibitively expensive when there are very many answers (Figure 2, left), which is why we do not produce them all by default.)

In more detail, adding conflicts to the ABox complicates the explanations of answers, due to their shift from the Sure to the Likely and Possible classes, as Table 3 shows. Explaining such answers indeed comes at higher computational cost. Figure 2 illustrates this phenomenon. The general trend is exemplified with $q3$: adding more conflicts causes the difficulty of explaining to grow more rapidly.

For negative IAR-answers, we compared the generation of explanations using a polynomial procedure and of *smallest* explanations using the SAT solver. The sizes of arbitrary explanations are generally very close to those of smallest explanations (at most two extra assertions on $c29$) and the time saved may be important. Arbitrary explanations are generated in less than 10ms, whereas for $g2$ on $c29$, around 50 minutes is spent in computing a smallest explanation due to its unusual size (18 assertions, whereas other negative explanations consist in a few assertions).

Finally, we observed that the average number of explanations per answer is often reasonably low, although some answers have a large number of explanations (e.g. on $c29$, often less than 10 on average, but for $g2$, 686 for an IAR-answer, 4210 for an AR-answer, and 740 for a brave-answer). Regarding the size of explanations of AR-answers, the number of causes in the disjunction was up to 21, showing the practical interest of ranking the explanations.

| | g2 | | | g3 | | | q1 | | | q2 | | | q3 | | | q4 | | |
|----------|----|------|-----|-----|------|-----|-----|-----|-----|----|-----|-----|----|-----|-----|-------|------|------|
| | c5 | c29 | c44 | c5 | c29 | c44 | c5 | c29 | c44 | c5 | c29 | c44 | c5 | c29 | c44 | c5 | c29 | c44 |
| Sure | 95 | 60 | 38 | 99 | 82 | 61 | 100 | 0 | 0 | 29 | 0 | 0 | 27 | 0 | 0 | 84.85 | 66.4 | 50.1 |
| Likely | 2 | 2.4 | 4 | 0.2 | 5.5 | 12 | 0 | 100 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.06 | 0.1 | 0.6 |
| Possible | 3 | 37.6 | 58 | 0.8 | 12.5 | 27 | 0 | 0 | 100 | 71 | 100 | 100 | 73 | 100 | 100 | 15.09 | 33.5 | 49.3 |

Table 3: Distribution of answers in the different classes on ABoxes with 5%, 29%, and 44% of assertions involved in a conflict.

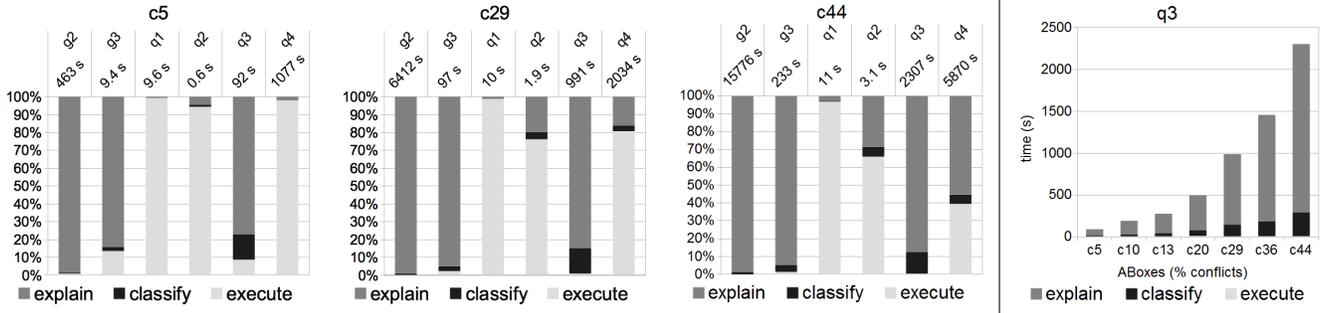


Figure 2: Impact of conflicts [left] Proportion of time spent in the different phases and total time (in sec.) to explain all query answers on c5, c29, and c44. [right] Time spent for explaining all answers of q3 on ABoxes with growing ratio of conflicts.

6 Related Work on Explanations

As mentioned in Section 1, there has been significant interest in equipping DL reasoning systems with explanation facilities. The earliest work proposed formal proof systems as a basis for explaining concept subsumptions (McGuinness and Borgida 1995; Borgida, Franconi, and Horrocks 2000), while the post-2000 literature mainly focuses on *axiom pinpointing* (Schlobach and Cornet 2003; Kalyanpur et al. 2005; Horridge, Parsia, and Sattler 2012), in which the problem is to generate minimal subsets of the KB that yield a given (surprising or undesirable) consequence; such subsets are often called *justifications*. For the lightweight DL \mathcal{EL}^+ , justifications correspond to minimal models of propositional Horn formulas and can be computed using SAT solvers (Sebastiani and Vescovi 2009); a polynomial algorithm has been proposed to compute one justification in (Baader, Peñaloza, and Suntisrivaraporn 2007). In DL-Lite, the problem is simpler: all justifications can be enumerated in polynomial delay (Peñaloza and Sertkaya 2010).

It should be noted that work on axiom pinpointing has thus far focused on explaining entailed TBox axioms (or possibly ABox assertions), but not answers to conjunctive queries. The latter problem is considered in (Borgida, Calvanese, and Rodriguez-Muro 2008), which introduces a proof-theoretic approach to explaining positive answers to CQs over DL-Lite_A KBs. The approach outputs a single proof, involving both TBox axioms and ABox assertions, using minimality criteria to select a ‘simplest’ proof.

More recently, the problem of explaining negative query answers over DL-Lite_A KBs has been studied (Calvanese et al. 2013). Formally, the explanations for $\mathcal{T}, \mathcal{A} \not\models q(\vec{a})$ correspond to sets \mathcal{A}' of ABox assertions such that $\mathcal{T}, \mathcal{A} \cup \mathcal{A}' \models q(\vec{a})$. Practical algorithms and an implementation for computing such explanations were described in (Du, Wang, and Shen 2014). The latter work was recently extended to the case of inconsistent KBs (Du, Wang, and Shen 2015). Es-

entially the idea is to add a set of ABox assertions that will lead to the answer holding under IAR semantics (in particular, the new assertions must not introduce any inconsistencies). By contrast, in our setting, negative query answers result not from the absence of supporting facts, but rather the presence of conflicting assertions. This is why our explanations are composed of assertions from the original ABox.

Probably the closest related work is (Arioua, Tamani, and Croitoru 2015) which introduces an argumentation framework for explaining positive and negative answers under the inconsistency-tolerant semantics ICR (Bienvenu 2012). Their motivations are quite similar to our own, and there are some high-level similarities in the definition of explanations (e.g. to explain positive ICR-answers, they consider sets of arguments that minimally cover the preferred extensions, whereas for positive AR-answers, we use sets of causes that minimally cover the repairs). Our work differs from theirs by considering different semantics and by providing a detailed complexity analysis and implemented prototype.

Finally, we note that the problem of explaining query results has been studied in the database community (Cheney, Chiticariu, and Tan 2009; Herschel and Hernández 2010).

7 Conclusion and Future Work

We devised a framework for explaining query (non-)answers over DL KBs under three well-established inconsistency-tolerant semantics (brave, AR, IAR). We then studied the computational properties of our framework, focusing on DL-Lite_R that underpins W3C’s OWL2 QL (Motik et al. 2012). For intractable explanation tasks, we exhibited tight connections with variants of propositional satisfiability, enabling us to implement a prototype using modern SAT solvers. Our experiments showed its practical interest: explanations of query (non-)answers are generated quickly overall, for realistic to challenging ratios of conflicting assertions.

There are several natural directions for future work. First, we plan to accompany our explanations with details on the TBox reasoning involved, using the work of (Borgida, Calvanese, and Rodriguez-Muro 2008) on proofs of positive answers as a starting point. The difficulty of such proofs could provide an additional criteria for ranking explanations (cf. the work on the cognitive complexity of justifications (Horridge et al. 2011)). Second, our experiments showed that an answer can have a huge number of explanations, many of which are quite similar in structure. We thus plan to investigate ways of improving the presentation of explanations, e.g. by identifying and grouping similar explanations (cf. (Bail, Parsia, and Sattler 2013) on comparing justifications), or by defining a notion of representative explanation as in (Du, Wang, and Shen 2014). Third, we plan to experiment with other methods of generating explanations of negative answers, by comparing alternative encodings and using tools for computing hitting sets or diagnoses. Finally, it would be interesting to explore how explanations can be used to partially repair the data based upon the user’s feedback.

Acknowledgements

This work was supported by the ANR project PAGODA (ANR-12-JS02-007-01).

References

- Arioua, A.; Tamani, N.; Croitoru, M.; and Buche, P. 2014. Query failure explanation in inconsistent knowledge bases using argumentation. In *Proc. of COMMA*.
- Arioua, A.; Tamani, N.; and Croitoru, M. 2015. Query answering explanation in inconsistent datalog +/- knowledge bases. In *Proc. of DEXA*.
- Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D.; and Patel-Schneider, P. F., eds. 2003. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press.
- Baader, F.; Peñaloza, R.; and Suntisrivaraporn, B. 2007. Pinpointing in the description logic EL^+ . In *Proc. of KI*.
- Bail, S.; Parsia, B.; and Sattler, U. 2013. The logical diversity of explanations in OWL ontologies. In *Proc. of CIKM*.
- Berre, D. L., and Parrain, A. 2010. The sat4j library, release 2.2. *JSAT* 7(2-3):59–64.
- Bertossi, L. E. 2011. *Database Repairing and Consistent Query Answering*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers.
- Bienvenu, M., and Rosati, R. 2013. Tractable approximations of consistent query answering for robust ontology-based data access. In *Proc. of IJCAI*.
- Bienvenu, M.; Bourgaux, C.; and Goasdoué, F. 2014. Querying inconsistent description logic knowledge bases under preferred repair semantics. In *Proc. of AAAI*.
- Bienvenu, M. 2012. On the complexity of consistent query answering in the presence of simple ontologies. In *Proc. of AAAI*.
- Borgida, A.; Calvanese, D.; and Rodriguez-Muro, M. 2008. Explanation in the DL-Lite family of description logics. In *Proc. of OTM*.
- Borgida, A.; Franconi, E.; and Horrocks, I. 2000. Explaining ALC subsumption. In *Proc. of ECAI*.
- Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2007. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. Autom. Reasoning (JAR)* 39(3):385–429.
- Calvanese, D.; Ortiz, M.; Simkus, M.; and Stefanoni, G. 2013. Reasoning about explanations for negative query answers in DL-Lite. *J. Artif. Intell. Res. (JAIR)* 48:635–669.
- Cheney, J.; Chiticariu, L.; and Tan, W. C. 2009. Provenance in databases: Why, how, and where. *Foundations and Trends in Databases* 1(4):379–474.
- Du, J.; Wang, K.; and Shen, Y. 2014. A tractable approach to abox abduction over description logic ontologies. In *Proc. of AAAI*.
- Du, J.; Wang, K.; and Shen, Y. 2015. Towards tractable and practical abox abduction over inconsistent description logic ontologies. In *Proc. of AAAI*.
- Herschel, M., and Hernández, M. A. 2010. Explaining missing answers to SPJUA queries. *PVLDB* 3(1):185–196.
- Horridge, M.; Bail, S.; Parsia, B.; and Sattler, U. 2011. The cognitive complexity of OWL justifications. In *Proc. of ISWC*.
- Horridge, M.; Parsia, B.; and Sattler, U. 2012. Extracting justifications from biportal ontologies. In *Proc. of ISWC*.
- Kalyanpur, A.; Parsia, B.; Sirin, E.; and Hendler, J. A. 2005. Debugging unsatisfiable classes in OWL ontologies. *J. Web Sem.* 3(4):268–293.
- Lembo, D.; Lenzerini, M.; Rosati, R.; Ruzzi, M.; and Savo, D. F. 2010. Inconsistency-tolerant semantics for description logics. In *Proc. of RR*.
- Liberatore, P. 2005. Redundancy in logic I: Cnf propositional formulae. *Artif. Intell. (AIJ)*.
- Lukasiewicz, T.; Martinez, M. V.; and Simari, G. I. 2013. Complexity of inconsistency-tolerant query answering in datalog+/- . In *Proc. of OTM*, 488–500.
- Lutz, C.; Seylan, I.; Toman, D.; and Wolter, F. 2013. The combined approach to OBDA: Taming role hierarchies using filters. In *Proc. of ISWC*.
- McGuinness, D. L., and Borgida, A. 1995. Explaining subsumption in description logics. In *Proc. of IJCAI*.
- Motik, B.; Cuenca Grau, B.; Horrocks, I.; Wu, Z.; Fokoue, A.; and Lutz, C. 2012. OWL 2 Web Ontology Language profiles. W3C Recommendation. Available at <http://www.w3.org/TR/owl2-profiles/>.
- Peñaloza, R., and Sertkaya, B. 2010. Complexity of axiom pinpointing in the dl-lite family of description logics. In *Proc. of ECAI*.
- Rosati, R.; Ruzzi, M.; Graziosi, M.; and Masotti, G. 2012. Evaluation of techniques for inconsistency handling in OWL 2 QL ontologies. In *Proc. of ISWC*.
- Rosati, R. 2011. On the complexity of dealing with inconsistency in description logic ontologies. In *Proc. of IJCAI*.
- Schlobach, S., and Cornet, R. 2003. Non-standard reasoning services for the debugging of description logic terminologies. In *Proc. of IJCAI*.
- Sebastiani, R., and Vescovi, M. 2009. Axiom pinpointing in lightweight description logics via horn-sat encoding and conflict analysis. In *Proc. of CADE*.

A Proofs for Section 4

We recall the definitions of the considered complexity classes:

- P: problems which are solvable in polynomial time.
- NP: problems which are solvable in non-deterministic polynomial time.
- coNP: problems whose complement is in NP.
- BH₂: problems that are the intersection of a problem in NP and a problem in coNP.
- Σ_2^p : problems which are solvable in non-deterministic polynomial time with an NP oracle.
- Π_2^p : problems whose complement is in Σ_2^p .

When showing that a decision problem is hard for a given complexity class, we use standard polynomial-time many-one reductions (also known as Karp reductions), which transform an instance of one decision problem into an instance of a second decision problem.

We consider that a procedure solves the generation task GENONE (resp. GENBEST) if it outputs an explanation (resp. best explanation according to the chosen criterion) when there is at least one explanation, and otherwise, it outputs no. To show that a generation task is hard for a class \mathcal{C} , we reduce a \mathcal{C} -hard decision problem to it. As we cannot use many-one reductions (which relate two decision problems), we will use polynomial-time Turing reductions, that is, we will show how to solve the \mathcal{C} -hard decision problem using a polynomial-time Turing machine that can use the generation task as an oracle. Moreover, to prove a stronger intractability result, we will only allow a single oracle call.

Positive brave and IAR-answers

In DL-Lite_R, the conflicts for a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ can be computed in polynomial time in $|\mathcal{A}|$ by querying (the database corresponding to) \mathcal{A} for individuals belonging to two disjoint concepts. More precisely, we can use the query obtained by applying standard UCQ rewriting algorithms to the queries corresponding to the violation of negative inclusions of the TBox. The conflicts are those sets of (at most two) assertions that correspond to the homomorphic image of one of the disjuncts of this query. The causes of q w.r.t. \mathcal{K} can be computed similarly: we first compute the sets of assertions that correspond to the homomorphic image of one of the disjuncts in a UCQ-rewriting of the query, and then prune the inconsistent or non-minimal sets.

The explanations for positive brave-answers are the causes, and the explanations for positive IAR-answers are the causes that belong to every repair, i.e., those which do not contain any assertion involved in a conflict of \mathcal{K} (recall that in DL-Lite_R conflicts are binary). It is therefore possible to compute the entire set of explanations for positive brave and IAR-answers in P. This means that GENONE is in P. For polynomial-time ranking criteria, both GENBEST and BEST REC are solvable in P since we can compare all of the explanations to identify the best ones. The sets of relevant and necessary assertions can be computed in P by taking the union and intersection of the explanations.

Positive AR-answers

We begin by proving Proposition 1. For convenience, we recall the definition of the clause sets:

$$\varphi_{-q} = \{\lambda_C \mid C \in \text{causes}(q(\vec{a}), \mathcal{K})\} \text{ with } \lambda_C = \bigvee_{\beta \in \text{confl}(C, \mathcal{K})} x_\beta$$

$$\varphi_{\text{cons}} = \{\neg x_\alpha \vee \neg x_\beta \mid x_\alpha, x_\beta \in \text{vars}(\varphi_{-q}), \{\alpha, \beta\} \in \text{confl}(\mathcal{K})\}$$

Proposition 1. A set $\mathcal{E} \subseteq \text{causes}(q(\vec{a}), \mathcal{K})$ is an explanation for $\mathcal{K} \models_{\text{AR}} q(\vec{a})$ iff $\{\lambda_C \mid C \in \mathcal{E}\}$ is a MUS of φ_{-q} w.r.t. φ_{cons} .

Proof. First suppose that \mathcal{E} is an explanation of $\mathcal{K} \models_{\text{AR}} q(\vec{a})$. By Definition 2, this means that every repair of \mathcal{K} contains at least one cause C from \mathcal{E} . It follows that it is not possible to select one conflicting assertion for each corresponding cause in a consistent way, i.e. $\{\lambda_C \mid C \in \mathcal{E}\} \cup \varphi_{\text{cons}}$ is inconsistent. Moreover, the minimality condition ensures that for every proper subset $\mathcal{E}' \subsetneq \mathcal{E}$, there is a repair \mathcal{R} that does not contain any cause from \mathcal{E}' . We can use \mathcal{R} to select a consistent set of assertions that conflict with every cause in \mathcal{E}' , which means that $\{\lambda_C \mid C \in \mathcal{E}'\} \cup \varphi_{\text{cons}}$ is satisfiable. Thus, $\{\lambda_C \mid C \in \mathcal{E}\}$ is a MUS of φ_{-q} w.r.t. φ_{cons} .

Conversely, suppose that $\{\lambda_C \mid C \in \mathcal{E}\}$ is a MUS of φ_{-q} w.r.t. φ_{cons} . Then $\{\lambda_C \mid C \in \mathcal{E}\} \cup \varphi_{\text{cons}}$ is unsatisfiable, and every $\{\lambda_C \mid C \in \mathcal{E}'\} \cup \varphi_{\text{cons}}$ with $\mathcal{E}' \subsetneq \mathcal{E}$ is satisfiable. The fact that $\{\lambda_C \mid C \in \mathcal{E}\} \cup \varphi_{\text{cons}}$ is unsatisfiable means that there is no way to consistently contradict the causes in \mathcal{E} , so every repair must contain one of the causes in \mathcal{E} . The satisfiability of $\{\lambda_C \mid C \in \mathcal{E}'\} \cup \varphi_{\text{cons}}$ for $\mathcal{E}' \subsetneq \mathcal{E}$ implies the existence of a repair that omits every cause in \mathcal{E}' . We have thus shown that \mathcal{E} satisfies the conditions of Definition 2, so it is an explanation of $\mathcal{K} \models_{\text{AR}} q(\vec{a})$. \square

Using Proposition 1 and known complexity results for MUSes yields the following upper bounds:

Proposition 4. *Regarding explanations for AR-answers, REC is in BH₂, BEST REC is in Π_2^p , and REL is in Σ_2^p w.r.t. data complexity.*

Proof. We recall the following complexity results for MUSes (see (Liberatore 2005)):

- Deciding if a set of clauses is a MUS is BH₂-complete.
- Deciding if a clause belongs to some MUS is Σ_2^p -complete.

When combined with Proposition 1, the first item yields membership in BH₂ of REC. For BEST REC, we show that an explanation is *not* a best one by guessing a better candidate and checking in BH₂ that it is an explanation. This yields a Σ_2^p procedure for the complement of BEST REC, hence membership in Π_2^p for BEST REC.

For REL, we note that an assertion α is relevant for explaining $\mathcal{K} \models_{\text{AR}} q(\vec{a})$ just in the case that there exists a cause C for $q(\vec{a})$ w.r.t. \mathcal{K} that contains α and appears in some explanation. By Proposition 1, the latter holds just in the case that λ_C belongs to some MUS of φ_{-q} w.r.t. φ_{cons} . By the second item above, deciding whether a particular clause λ_C belongs to some MUS can be decided in Σ_2^p . To obtain a Σ_2^p decision procedure for REL, we simply add an initial non-deterministic guess of a cause $C \in \text{causes}(q(\vec{a}), \mathcal{K})$ that mentions the considered assertion α . \square

We next show the NP upper bound for NEC.

Proposition 5. *Regarding explanations for AR-answers, NEC is in NP w.r.t. data complexity.*

Proof. An assertion α belongs to every explanation of $\mathcal{K} \models_{\text{AR}} q(\vec{a})$ just in the case that either there are no explanations at all (i.e., $\mathcal{K} \not\models_{\text{AR}} q(\vec{a})$) or there exists a repair \mathcal{R} of $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ such that $\mathcal{T}, \mathcal{R} \setminus \{\alpha\} \not\models q(\vec{a})$. Both conditions can be tested in NP w.r.t. data complexity. Indeed, to decide whether the second condition holds, we simply guess a subset $\mathcal{R} \subseteq \mathcal{A}$ and check (in P w.r.t. data complexity) that \mathcal{R} is a repair and $\mathcal{T}, \mathcal{R} \setminus \{\alpha\} \not\models q(\vec{a})$. \square

The following proposition shows how the connection to MUSes can be exploited to obtain matching lower bounds.

Proposition 6. *Regarding explanations for AR-answers, REC is BH₂-hard, NEC is NP-hard, REL is Σ_2^p -hard, and GENONE is NP-hard w.r.t. data complexity. Moreover, if we rank explanations according to the number of causes or number of assertions, then BEST REC (resp. GENBEST) is Π_2^p -hard (resp. Σ_2^p -hard) w.r.t. data complexity. .*

Proof. We show how the MUSes of a propositional clause set can be captured by explanations of AR-answers.

Let $\varphi_0 = \{C_1, \dots, C_n\}$ be a set of clauses over $\{X_1, \dots, X_p\}$. Consider the following KB and query (borrowed from (Bienvenu 2012)):

$$\begin{aligned} \mathcal{T}_0 &= \{\exists P^- \sqsubseteq \neg \exists N^-, \exists U^- \sqsubseteq \neg \exists P, \exists U^- \sqsubseteq \neg \exists N, \\ &\quad \exists U \sqsubseteq A\} \\ \mathcal{A}_0 &= \{P(c_i, x_j) \mid X_j \in C_i\} \cup \{N(c_i, x_j) \mid \neg X_j \in C_i\} \cup \\ &\quad \{U(a, c_i) \mid 1 \leq i \leq n\} \\ q_0 &= A(x) \end{aligned}$$

The causes for $q_0(a)$ are given by the assertions $U(a, c_i)$, which are in conflict with assertions of the form $P(c_i, x_j)$ or $N(c_i, x_j)$. It was shown in (Bienvenu 2012) that $\mathcal{T}_0, \mathcal{A}_0 \models_{\text{AR}} A(a)$ iff φ_0 is unsatisfiable. To prove the proposition, we will require the following stronger claim:

Claim. The following are equivalent:

1. the set of clauses $\{C_{i_1}, \dots, C_{i_k}\}$ is unsatisfiable
2. every repair of $(\mathcal{T}_0, \mathcal{A}_0)$ contains some assertion from $\{U(a, c_{i_1}), \dots, U(a, c_{i_k})\}$

Proof of claim. It will be more convenient to show that the negations of the two statements are equivalent. First suppose that $\{C_{i_1}, \dots, C_{i_k}\}$ is satisfiable, as witnessed by the satisfying assignment ν . Define a repair \mathcal{R}_ν of $(\mathcal{T}_0, \mathcal{A}_0)$ by including the assertion $P(c_i, v_j)$ if $\nu(v_j) = \text{true}$, including $N(c_i, v_j)$ if $\nu(v_j) = \text{false}$, and then adding as many other assertions as needed to obtain a maximal \mathcal{T}_0 -consistent subset. Since ν satisfies every clause in $\{C_{i_1}, \dots, C_{i_k}\}$, it follows that for every index $\ell \in \{i_1, \dots, i_k\}$, the clause C_ℓ contains a positive literal v_ℓ such that $\nu(v_\ell) = \text{true}$, or a negative literal $\neg v_\ell$ such that $\nu(v_\ell) = \text{false}$. In the former case, \mathcal{R}_ν contains the assertion $P(c_\ell, v_\ell)$, and in the latter case, \mathcal{R}_ν contains $N(c_\ell, v_\ell)$. In both cases, there is an assertion in \mathcal{R}_ν that conflicts with $U(a, c_\ell)$, so the latter assertion cannot appear in \mathcal{R}_ν . We have thus shown that \mathcal{R}_ν does not contain any of the assertions in $\{U(a, c_{i_1}), \dots, U(a, c_{i_k})\}$.

Next suppose there is a repair \mathcal{R} that have an empty intersection with $\{U(a, c_{i_1}), \dots, U(a, c_{i_k})\}$. By the maximality of \mathcal{R} , it follows that for every $\ell \in \{i_1, \dots, i_k\}$, there must exist an assertion in \mathcal{R} of the form $P(c_\ell, v_\ell)$ or $N(c_\ell, v_\ell)$. Define a (possibly partial) assignment $\nu_\mathcal{R}$ by setting by X_j to true if \mathcal{R} contains some $P(c_i, x_j)$ and to false if \mathcal{R} contains some $N(c_i, x_j)$ (recall that \mathcal{R} is consistent with \mathcal{T}_0 , and so it cannot contain both $P(c_i, x_j)$ and $N(c_i, x_j)$). By construction, $\nu_\mathcal{R}$ satisfies all of the clauses in $\{C_{i_1}, \dots, C_{i_k}\}$, i.e. $\{C_{i_1}, \dots, C_{i_k}\}$ is satisfiable. (*end proof of claim*)

It follows from the preceding claim that the explanations for $\mathcal{T}_0, \mathcal{A}_0 \models_{\text{AR}} q_0(a)$, i.e. the minimal sets of causes for $q_0(a)$ that cover all repairs, correspond precisely to the MUSes of φ_0 . We can therefore exploit known complexity results for MUSes (Liberatore 2005):

- Deciding if a clause belongs to a MUS is Σ_2^p -complete, so deciding if $U(a, c_i)$ belongs to an explanation for $\mathcal{T}_0, \mathcal{A}_0 \models_{\text{AR}} q_0(a)$ is Σ_2^p -hard w.r.t. data complexity. Thus, we have a Σ_2^p lower bound for REL.
- Deciding if a clause belongs to every MUS is NP-complete, so deciding if $U(a, c_i)$ belongs to every explanation for $\mathcal{T}_0, \mathcal{A}_0 \models_{\text{AR}} q_0(a)$ is NP-hard w.r.t. data complexity. This gives an NP lower bound for NEC.
- REC: Deciding if a set of clauses is a MUS is BH_2 -complete, so deciding if $\{\{U(a, c_{i_1})\}, \dots, \{U(a, c_{i_k})\}\}$ is an explanation is BH_2 -hard w.r.t. data complexity. Hence, REC is BH_2 -hard.

The proof of (Liberatore 2005) for Σ_2^p -hardness of deciding if there exists a MUS of size at most k also shows that deciding if a set of clauses is a smallest MUS is Π_2^p -hard. It follows that deciding if an explanation for an AR-answer contains a smallest number of causes is Π_2^p -hard. Moreover, since every cause in the considered KB consists of a single assertion, deciding if an explanation for an AR-answer contains a smallest number of assertions is also Π_2^p -hard.

To see why the generation task GENONE is NP-hard, we note that to solve the NP-complete problem of whether $\mathcal{K} \not\models_{\text{AR}} q(\vec{a})$, it suffices to call the procedure for GENONE to generate a single explanation for $\mathcal{K} \models_{\text{AR}} q(\vec{a})$. If the procedure outputs ‘no’ (meaning there is no explanation for $\mathcal{K} \models_{\text{AR}} q(\vec{a})$), then we output ‘yes’, and if it outputs an explanation, then we return ‘no’.

The Σ_2^p -hardness of GENBEST, when explanations are ranked based upon the number of disjuncts or the number of assertions, follows from the Π_2^p -hardness of BEST REC for these same criteria. Indeed, to show that an explanation is *not* a best explanation, it suffices to generate a best explanation (GENBEST) and verify that it has fewer disjuncts / assertions than the explanation at hand. \square

Negative AR-answers

We begin by establishing Proposition 2.

Proposition 2. A set \mathcal{E} is an explanation (resp. cardinality-minimal explanation) for $\mathcal{K} \not\models_{\text{AR}} q(\vec{a})$ iff $\{x_\alpha \mid \alpha \in \mathcal{E}\}$ is a minimal (resp. cardinality-minimal) model of $\varphi_{\neg q} \cup \varphi_{\text{cons}}$.

Proof. It is shown in (Bienvenu, Bourgaux, and Goasdoué 2014) that $\mathcal{K} \models_{\text{AR}} q(\vec{a})$ iff $\varphi_{\neg q} \wedge \varphi_{\text{cons}}$ is unsatisfiable. This is because the assertions whose corresponding variables are assigned to true in a valuation that satisfies $\varphi_{\neg q} \wedge \varphi_{\text{cons}}$ form a subset of the ABox which: (i) contradicts every cause, since $\varphi_{\neg q}$ states that for every cause, one conflicting assertion is selected, and (ii) is consistent, since φ_{cons} states that two assertions in a conflict cannot be selected together. Thus, the inclusion-minimal models of $\varphi_{\neg q} \wedge \varphi_{\text{cons}}$ are precisely the explanations for negative AR-answers. \square

Next we show the complexity upper bounds for the decision problems.

Proposition 7. *Regarding explanations for negative AR-answers, REC is in P, BEST REC is in coNP, REL is in NP, and NEC is in coNP w.r.t. data complexity.*

Proof. It follows from Definition 3 that deciding whether $\mathcal{E} \subseteq \mathcal{A}$ is an explanation for $\mathcal{K} \not\models_{\text{AR}} q(\vec{a})$ can be done in P (data complexity) by checking:

- consistency of $(\mathcal{T}, \mathcal{E})$
- inconsistency of $(\mathcal{T}, \mathcal{E} \cup \mathcal{C})$ for every $\mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K})$
- minimality of \mathcal{E} : no proper subset $\mathcal{E}' \subsetneq \mathcal{E}$ satisfies the two previous conditions.

We can decide in NP that an explanation \mathcal{E} is *not* a best explanation (according to some polynomial-time ranking criterion) by guessing a subset $\mathcal{E}' \subseteq \mathcal{A}$ and verifying in P w.r.t. data complexity that \mathcal{E}' is an explanation (see previous paragraph) and that it is better than \mathcal{E} according to the given criterion. This yields a coNP upper bound for BEST REC.

A simple NP procedure for deciding REL consists in guessing a subset $\mathcal{E} \subseteq \mathcal{A}$ that contains the considered assertion and checking in P whether it is an explanation (using the P procedure for REC).

However, for the purposes of implementation, we propose an alternative procedure based upon a reduction to satisfiability. Specifically, to test whether an assertion α is relevant, we check whether the clause set $\varphi_{\neg q} \cup \varphi_{\text{cons}} \cup \varphi_\alpha$ is satisfiable, where

$$\varphi_\alpha = \left\{ \bigvee_{\mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K}), \alpha \in \text{confl}(\mathcal{C}, \mathcal{K})} x_{\mathcal{C}} \right\} \cup \left\{ \neg x_{\mathcal{C}} \vee \neg x_{\beta} \mid \mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K}), \alpha \in \text{confl}(\mathcal{C}, \mathcal{K}), \beta \in \text{confl}(\mathcal{C}, \mathcal{K}), \beta \neq \alpha \right\}$$

Indeed, if α is relevant, there exists an explanation \mathcal{E} such that $\alpha \in \mathcal{E}$. Since \mathcal{E} is minimal, there exists a cause \mathcal{C} such that $\mathcal{C} \cup (\mathcal{E} \setminus \{\alpha\})$ is consistent. It follows that no assertion $\beta \in \text{confl}(\mathcal{C}, \mathcal{K})$ belongs to \mathcal{E} expect for α . Then the valuation ν such that $\nu(x_{\mathcal{C}}) = \text{true}$, and for every assertion β , $\nu(x_{\beta}) = \text{true}$ if $\beta \in \mathcal{E}$, $\nu(x_{\beta}) = \text{false}$ otherwise, satisfies $\varphi_{\neg q} \cup \varphi_{\text{cons}} \cup \varphi_\alpha$. In the other direction, if $\varphi_{\neg q} \cup \varphi_{\text{cons}} \cup \varphi_\alpha$ is satisfiable, it is possible to contradict every cause with a consistent set \mathcal{E} of assertions such that there exists a cause \mathcal{C} such that the only assertion of $\mathcal{E} \cap \text{confl}(\mathcal{C}, \mathcal{K})$ is α . Then an explanation that contains α is included in \mathcal{E} .

By Proposition 2, \mathcal{E} is an explanation for $\mathcal{K} \not\models_{\text{AR}} q(\vec{a})$ iff $\{x_\alpha \mid \alpha \in \mathcal{E}\}$ is a minimal model of $\varphi_{\neg q} \cup \varphi_{\text{cons}}$. It follows that an assertion α belongs to every explanation for $\mathcal{K} \not\models_{\text{AR}} q(\vec{a})$ just in the case that $\varphi_{\neg q} \cup \varphi_{\text{cons}} \cup \{\neg x_\alpha\}$ is unsatisfiable. This yields membership in coNP for NEC. \square

The next proposition establishes matching lower bounds.

Proposition 8. *Regarding explanations for negative AR-answers, NEC is coNP-hard, and REL, GENONE, and GENBEST (for any ranking criterion) is NP-hard w.r.t. data complexity. If explanations are ranked by cardinality, then BEST REC is coNP-hard w.r.t. data complexity.*

Proof. All reductions are from (UN)SAT. Let $\varphi = C_1 \wedge \dots \wedge C_n$ be a set of clauses over propositional variables $\{X_1, \dots, X_p\}$.

• GENONE and GENBEST: Let $\mathcal{T}_0, \mathcal{A}_0$, and q_0 be as in Proposition 6. We know that φ_0 is satisfiable iff $\mathcal{T}_0, \mathcal{A}_0 \not\models_{\text{AR}} A(a)$. Thus, to decide the satisfiability of φ_0 , we generate a (best) explanation of $\mathcal{T}_0, \mathcal{A}_0 \not\models_{\text{AR}} A(a)$. If an explanation is produced, then we return ‘yes’, and if the procedure returns with no explanation, then we output ‘no’.

• NEC: We again let $\mathcal{T}_0, \mathcal{A}_0$, and q_0 be as in Proposition 6. Define a new TBox $\mathcal{T}_1 = \mathcal{T}_0 \cup \{\exists U \sqsubseteq \neg S\}$ and ABox $\mathcal{A}_1 = \mathcal{A}_0 \cup \{S(a)\}$. By construction, the assertion $S(a)$ contradicts every cause for $q_0(a)$, so $\mathcal{T}_1, \mathcal{A}_1 \not\models_{\text{AR}} q_0(a)$. We show that deciding whether φ is satisfiable is equivalent to deciding if $S(a)$ is *not* necessary for explaining $\mathcal{T}_1, \mathcal{A}_1 \not\models_{\text{AR}} q_0(a)$. This establishes the coNP-hardness of checking necessity.

\Rightarrow Let ν be a satisfying valuation for φ . It can be easily verified that the set $\{P(c_i, v_j) \in \mathcal{A}_0 \mid \nu(v_j) = \text{true}\} \cup \{N(c_i, v_j) \in \mathcal{A}_0 \mid \nu(v_j) = \text{false}\}$ conflicts with every cause of $q_0(a)$, and so by choosing a subset of these assertions, we can construct an explanation for $\mathcal{T}_1, \mathcal{A}_1 \not\models_{\text{AR}} q_0(a)$ that does not contain $S(a)$.

\Leftarrow An explanation \mathcal{E} that does not contain $S(a)$ forms a \mathcal{T}_1 -consistent set of P - and N -assertions such that every c_i has an outgoing P - or N -edge. We obtain a (partial) assignment $\nu_{\mathcal{E}}$ that satisfies φ by setting $\nu_{\mathcal{E}}(v_j) = \text{true}$ if \mathcal{E} contains an assertion $P(c_i, v_j)$ and $\nu_{\mathcal{E}}(v_j) = \text{false}$ if \mathcal{E} contains an assertion $N(c_i, v_j)$.

• REL: We use the TBox \mathcal{T}_1 and the ABox $\mathcal{A}_2 = \mathcal{A}_1 \cup \{U(a, c_{n+1}), P(c_{n+1}, x_{p+1})\}$. Again, we note that $S(a)$ contradicts every cause for $q_0(a)$, so $\mathcal{T}_1, \mathcal{A}_2 \not\models_{\text{AR}} q_0(a)$. We show that φ is satisfiable iff $P(c_{n+1}, x_{p+1})$ is relevant for explaining $\mathcal{T}_1, \mathcal{A}_2 \not\models_{\text{AR}} q_0(a)$; it follows that relevance is NP-hard.

\Rightarrow If φ is satisfiable, then we can obtain an explanation for $\mathcal{T}_1, \mathcal{A}_2 \not\models_{\text{AR}} q_0(a)$ by adding $P(c_{n+1}, x_{p+1})$ to a minimal subset of the P - and N -assertions corresponding to a satisfying truth assignment for φ .

\Leftarrow If φ is unsatisfiable, then every explanation must contain $S(a)$. It follows that $\{S(a)\}$ is the only explanation, so $P(c_{n+1}, x_{p+1})$ is not relevant.

• BEST REC: We consider the following KB:

$$\begin{aligned} \mathcal{T}_3 &= \mathcal{T}_0 \cup \{U_1 \sqsubseteq U, U_2 \sqsubseteq U, \exists U_1^- \sqsubseteq \neg T, \exists U_2 \sqsubseteq \neg S\} \\ \mathcal{A}_3 &= \{P(c_i, x_j) \mid X_j \in C_i\} \cup \{N(c_i, x_j) \mid \neg X_j \in C_i\} \cup \\ &\quad \{U_1(a, c_i), U_2(a, c_i), T(c_i) \mid 1 \leq i \leq n\} \cup \{S(a)\} \end{aligned}$$

We claim that $\mathcal{E} = \{S(a), T(c_1), \dots, T(c_m)\}$ is a smallest explanation for $\mathcal{T}_3, \mathcal{A}_3 \not\models_{\text{AR}} q_0(a)$ iff φ is unsatisfiable.

\Rightarrow If φ is satisfiable, then we can use a satisfying truth assignment to define a consistent set of m P - and N -edges such that every c_i has an outgoing edge. This set is an explanation for $\mathcal{T}_3, \mathcal{A}_3 \not\models_{\text{AR}} q_0(a)$, and it has fewer assertions than \mathcal{E} .

\Leftarrow If there exists an explanation of size at most m , it contains necessarily only P - and N -edges, since m assertions (P , N or T) are needed to conflict all U_1 , and $S(a)$ is needed as soon as one of the U_1 -assertions is conflicted only by a T -assertion. It follows that there exists a consistent set of P - and N -assertions such that every c_i has an outgoing edge, from which we can construct a satisfying assignment for φ . \square

Negative IAR-answers

We start by proving Proposition 3.

Proposition 3. A set \mathcal{E} is an explanation (resp. cardinality-minimal explanation) for $\mathcal{K} \not\models_{\text{IAR}} q(\vec{a})$ iff $\{x_\alpha \mid \alpha \in \mathcal{E}\}$ is a minimal (resp. cardinality-minimal) model of $\varphi_{\neg q}$.

Proof. The assertions whose corresponding variables are assigned to true in a valuation that satisfies $\varphi_{\neg q}$ form a subset of the ABox which contradicts every cause, since $\varphi_{\neg q}$ states that for every cause, one conflicting assertion is selected. Thus, the inclusion-minimal (resp. cardinality-minimal) models of $\varphi_{\neg q}$ are precisely the explanations (resp. cardinality-minimal explanations) for negative IAR-answers. \square

We next establish the complexity upper bounds.

Proposition 9. *Regarding explanations for negative IAR-answers, REC is in P, BEST REC is in coNP, NEC is in P, REL is in P, and GENONE is in P w.r.t. data complexity.*

Proof. It follows from Definition 3 and from the fact that in DL-Lite_R conflicts are binary that deciding whether $\mathcal{E} \subseteq \mathcal{A}$ is an explanation for $\mathcal{K} \not\models_{\text{IAR}} q(\vec{a})$ can be done in P (data complexity) by checking:

- for every $\mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K})$, inconsistency of $(\mathcal{T}, \mathcal{C} \cup \{\alpha\})$ for some assertion $\alpha \in \mathcal{E}$
- minimality of \mathcal{E} : no proper subset $\mathcal{E}' \subsetneq \mathcal{E}$ satisfies the previous condition.

We can decide in NP that an explanation \mathcal{E} is *not* a best explanation (according to some polynomial-time ranking criterion) by guessing a subset $\mathcal{E}' \subseteq \mathcal{A}$ and verifying in P w.r.t. data complexity that \mathcal{E}' is an explanation and that it is better than \mathcal{E} according to the given criterion. This yields a coNP upper bound for BEST REC.

An assertion is necessary just in the case that it is the only conflict of some cause. Since causes and conflicts can be computed in P, deciding whether an assertion is necessary can be done in P.

For REL and GENONE, we can use Proposition 3 to polynomially reduce these problems to the corresponding problems for minimal models of monotone CNF formulas and exploit known results for that setting. Here we describe polytime procedures for the REL and GENONE that are based upon standard techniques from the propositional setting.

The key property underlying the polynomial procedure for REL is as follows: an assertion α is relevant for $\mathcal{K} \not\models_{\text{IAR}} q(\vec{a})$ iff it is in conflict with a cause \mathcal{C} such that for every other cause \mathcal{C}' , if $\text{confl}(\mathcal{C}', \mathcal{K}) \subseteq \text{confl}(\mathcal{C}, \mathcal{K})$, then $\alpha \in \text{confl}(\mathcal{C}', \mathcal{K})$. Clearly, the latter condition can be checked in polynomial time by examining the causes and conflicts (which are known to be computable in P w.r.t. data complexity). To see why this characterization holds, first note that if α is relevant for $\mathcal{K} \not\models_{\text{IAR}} q(\vec{a})$, then there is a subset $\mathcal{E} \subseteq \mathcal{A}$ with $\alpha \in \mathcal{E}$ such that every cause of $q(\vec{a})$ is in conflict with some assertion in \mathcal{E} , and no proper subset of \mathcal{E} possesses this property. Since \mathcal{E} is a minimal set of assertions having this property, we know that there is some cause \mathcal{C} that does not conflict with any assertion in $\mathcal{E} \setminus \{\alpha\}$, and so there cannot exist another cause \mathcal{C}' such that $\text{confl}(\mathcal{C}', \mathcal{K}) \subseteq \text{confl}(\mathcal{C}, \mathcal{K})$ and $\alpha \notin \text{confl}(\mathcal{C}', \mathcal{K})$. Conversely, let us suppose that the assertion α is in conflict with a cause \mathcal{C} of $q(\vec{a})$ and for every other cause \mathcal{C}' , $\text{confl}(\mathcal{C}', \mathcal{K}) \subseteq \text{confl}(\mathcal{C}, \mathcal{K})$ implies $\alpha \in \text{confl}(\mathcal{C}', \mathcal{K})$. It follows that for every cause \mathcal{C}' of $q(\vec{a})$, either $\alpha \in \text{confl}(\mathcal{C}', \mathcal{K})$, or there exists an assertion $\beta_{\mathcal{C}'} \in \text{confl}(\mathcal{C}', \mathcal{K})$ such that $\beta_{\mathcal{C}'} \notin \text{confl}(\mathcal{C}, \mathcal{K})$. We can therefore construct an explanation for $\mathcal{K} \not\models_{\text{IAR}} q(\vec{a})$ by taking α together with some of the assertions $\beta_{\mathcal{C}'}$.

For GENONE, we first compute (in P) the set of causes of q and conflicts of \mathcal{K} . If there is some cause that does not participate in any conflict, then $\mathcal{K} \models_{\text{IAR}} q(\vec{a})$, so we return ‘no’. Otherwise, for each cause $\mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K})$, we choose some assertion $\alpha_{\mathcal{C}}$ such that $\alpha_{\mathcal{C}}$ conflicts with some assertion in \mathcal{C} . By construction, $\{\alpha_{\mathcal{C}} \mid \mathcal{C} \in \text{causes}(q(\vec{a}), \mathcal{K})\}$ contradicts all causes, which means that this set contains at least one explanation. We therefore proceed to remove one assertion at a time as long as the set retains the property of contradicting all causes. When it is no longer possible to remove any assertions, we return the current set of assertions, which is an explanation. \square

Finally, we establish the intractability of BEST REC and GENBEST.

Proposition 10. *Regarding explanations for negative IAR-answers in the case where explanations are ranked by cardinality, GENBEST is NP-hard, and BEST REC is coNP-hard w.r.t. data complexity.*

Proof. We give a reduction from the problem of deciding if a truth assignment that satisfies a monotone 2-SAT formula assigns a smallest number of variables to true. This problem is coNP-complete (coNP-hardness can be shown by a straightforward reduction from the complement of the well-known NP-complete vertex cover problem).

Let $\varphi = C_1 \wedge \dots \wedge C_n$ be a monotone 2-CNF over the variables $\{X_1, \dots, X_p\}$, and let ν be a truth assignment that satisfies φ . Consider the following KB:

$$\begin{aligned} \mathcal{T} &= \{\exists P_k^- \sqsubseteq \neg T \mid 1 \leq k \leq 2\} \\ \mathcal{A} &= \{T(x_i) \mid 1 \leq i \leq p\} \cup \{P_k(c_j, x_i) \mid X_i \text{ } k^{\text{th}} \text{ term of } C_j\} \\ q &= \exists y z_1 z_2 P_1(y, z_1) \wedge P_2(y, z_2) \end{aligned}$$

The causes for $q(a)$ take the form $\{P_1(c_j, x_{i_1}), P_2(c_j, x_{i_2})\}$. It follows that an explanation for $\mathcal{T}, \mathcal{A} \not\models_{\text{IAR}} q$ is a set \mathcal{E} of T -assertions such that for every c_j , there is at least one $X_i \in C_j$ such that $T(x_i) \in \mathcal{E}$.

Deciding if ν assigns a minimal number of variables to true is equivalent to deciding if $\mathcal{E} = \{T(x_i) \mid \nu(X_i) = \text{true}\}$ is a smallest explanation. This yields the coNP-hardness of BEST REC, as well as the NP-hardness of GENBEST: we can solve the minimum assignment problem - and its complement - by generating a cardinality-minimal explanation and comparing its size with the number of variables set to true by the candidate assignment. \square