Enhancing Controlled Query Evaluation through Epistemic Policies

Gianluca Cima[1] , **Domenico Lembo**[1] , **Lorenzo Marconi**[1] ,
Riccardo Rosati[1] and **Domenico Fabio Savo**[2]
[1]Sapienza University of Rome
[2]University of Bergamo
{lastname}@diag.uniroma1.it, domenicofabio.savo@unibg.it

Abstract

In this paper, we propose the use of epistemic dependencies to express data protection policies in Controlled Query Evaluation (CQE), which is a form of confidentiality-preserving query answering over ontologies and databases. The resulting policy language goes significantly beyond those proposed in the literature on CQE so far, allowing for very rich and practically interesting forms of data protection rules. We show the expressive abilities of our framework and study the data complexity of CQE for (unions of) conjunctive queries when ontologies are specified in the Description Logic DL-Lite$_{\mathcal{R}}$. Interestingly, while we show that the problem is in general intractable, we prove tractability for the case of acyclic epistemic dependencies by providing a suitable query rewriting algorithm. The latter result paves the way towards the implementation and practical application of this new approach to CQE.

1 Introduction

Controlled Query Evaluation (CQE) is a confidentiality-preserving query answering approach that protects sensitive information by filtering query answers in such a way that a user cannot deduce information declared confidential by a data protection policy [Biskup, 2000; Biskup and Bonatti, 2004; Bonatti and Sauro, 2013; Cuenca Grau *et al.*, 2013].

In CQE, one crucial aspect concerns the expressiveness of the policy language, which determines the form of the logic formulas definable in the policy, consequently influencing a designer's capacity to declare which pieces of information must not be disclosed. Previous literature has mainly considered only policies consisting of sentences, i.e. closed formulas, as in [Sicherman *et al.*, 1983; Biskup and Bonatti, 2004; Bonatti and Sauro, 2013; Lembo *et al.*, 2019]. Through this approach, it is only possible to impose that the truth value of a sentence in the policy cannot be inferred from the system by asking queries. For example, [Lembo *et al.*, 2019] study CQE when policy statements take the form $\forall\vec{x}(q(\vec{x}) \rightarrow \bot)$, referred to as denial. Enforcing one such denial over the data means refraining from disclosing the inference of the Boolean conjunctive query (BCQ) $\exists\vec{x}\ q(\vec{x})$ by the system, even if the inference has occurred. For instance, the rule

$$\delta_1 = \forall x, y(\mathsf{Patient}(x) \wedge \mathsf{admitted}(x, y) \rightarrow \bot)$$

says that it is confidential whether there exists a patient admitted in a hospital department. However, if the aim is to hide admitted patients, the above dependency is imposing an excessively stringent level of protection, since it is denying the presence of patients in the hospital to protect their identity, which is implausible in practice. A more effective approach would be instead to require that the system must not answer the *open* query $q(x) : \exists y(\mathsf{Patient}(x) \wedge \mathsf{admitted}(x, y))$. Intuitively, specifying a rule of this kind means imposing that the set of patients that the system *knows* to be admitted to the hospital has not to be disclosed.

To properly capture this behavior, we propose to use an epistemic operator K in policy formulas, in the spirit of [Calvanese *et al.*, 2007b; Console and Lenzerini, 2020]. This allows us to formalize the epistemic state of the user, that is, what the system can tell to the user without disclosing sensitive information. In our example, the policy rule is as follows:

$$\delta_2 = \forall x\big(K\,\exists y(\mathsf{Patient}(x) \wedge \mathsf{admitted}(x, y)) \rightarrow K\bot\big)$$

Rule δ_2 imposes that, in the epistemic state of the user, the set of admitted patients must be empty (but this does not exclude that the user knows that some patients have been admitted).

In fact, our proposal enables us to accomplish more than just that. In a more advanced scenario, concealing the relationship between a patient and a hospital should apply only if the patient has not signed a consensus form. This can be expressed by the following formula:

$$\delta_3 = \forall x\big(K\,\exists y(\mathsf{Patient}(x) \wedge \mathsf{admitted}(x, y)) \rightarrow K\mathsf{Consent}(x)\big)$$

Intuitively, the formula is saying that if a user knows that a patient has been admitted, then she must know that the patient has signed a consensus form. Thus, if a patient did not sign a consensus form, the system cannot disclose that this patient has been admitted. In general, this kind of policy is well-suited for encoding the principle of *privacy by default*, which is a desirable property in data protection (as expressly outlined in Article 25 of GDPR [European Union, 2016]).

We remark once again that policy rules of the form just described have not been previously considered in the literature. It is however worth noticing that in [Cuenca Grau *et al.*, 2015]

CQE is studied for a policy consisting of *one single open* CQ. In this latter framework, a rule analogous to δ_2 of our example can be in fact specified, but richer policies, as those requiring more denials and/or rules as δ_3 above, are non-expressible.

Formulas as δ_2 and δ_3 are called *epistemic dependencies* (EDs). EDs have been originally introduced in [Console and Lenzerini, 2020] to express integrity constraints in ontology-based data management and are indeed a special case of *domain-independent EQL-Lite(CQ)* sentences [Calvanese *et al.*, 2007a]. In the present paper, we use EDs as policy rules that must be satisfied to preserve data confidentiality over Description Logic (DL) ontologies. Similarly, integrity constraints must be satisfied to guarantee data consistency. However, our aim is totally different from that of [Console and Lenzerini, 2020], whose focus is on consistency checking.

After defining the policy language, to completely characterize our CQE framework, we need to specify its semantics. This issue is addressed in CQE through *censors*. In this paper, we use *CQ-censors* introduced in [Lembo *et al.*, 2019]. In a nutshell, given an ontology $\mathcal{O}$, an *optimal CQ-censor* is a maximal subset $\mathcal{C}$ of the set of all BCQs inferred by $\mathcal{O}$, such that $\mathcal{C}$ coupled with the intensional component of $\mathcal{O}$ (i.e. its TBox $\mathcal{T}$) satisfies all rules in the policy (i.e. no secrets are disclosed through standard query answering over $\mathcal{T} \cup \mathcal{C}$). As in [Lembo *et al.*, 2019], we then define CQE as the problem of computing the query answers that are in the intersection of the answer sets returned by all optimal censors. We call this problem *SC-entailment* (Skeptical entailment under CQ-censors). This form of CQE does not suffer from the problem of having to arbitrarily select an optimal censor among several incomparable ones (see e.g., in [Bonatti and Sauro, 2013; Cuenca Grau *et al.*, 2015]). CQ-censors are particularly interesting from a practical perspective, since, for ontologies specified DL-Lite$_\mathcal{R}$ [Calvanese *et al.*, 2007b] (a DL suited for modelling data-intensive ontologies) and policies expressed as denials, SC-entailment of BCQs is tractable in data complexity [Lembo *et al.*, 2019]. One of this paper's aims is then to study data complexity of SC-entailment of BCQs under epistemic policies for DL-Lite$_\mathcal{R}$ ontologies, with the ultimate goal of identifying conditions ensuring its tractability.

Besides the computational complexity study, we also carry out an analysis of the robustness of SC-entailment with respect to confidentiality-preservation. In [Biskup and Weibert, 2008; Bonatti and Sauro, 2013; Bonatti, 2022], it is shown that censoring mechanisms based on an indistinguishability criterion are indeed more secure than others. In abstract terms, according to such a criterion, confidentiality is guaranteed only if query answers returned over a data instance with sensitive information coincide with those returned over a data instance without secrets (which is thus indistinguishable from the other instance). In this paper, we investigate whether the entailment we consider enjoys indistinguishability.

Specifically, our main results are for ontologies in DL-Lite$_\mathcal{R}$ and policies that are sets of EDs. In more detail:

- As for indistinguishability, we show that SC-entailment of BCQs preserves confidentiality as defined in [Biskup and Weibert, 2008], but that this result does not carry over to unions of BCQs (BUCQs). We however prove that the property holds even for BUCQs in the case of IC-entailment, a sound approximation of SC-Entailment that considers a single censor given by the intersection of all optimal CQ-censors (Section 4);
- We show that SC-entailment and IC-entailment of BCQs and BUCQs are coNP-complete in data complexity (Section 5.1);
- For *acyclic EDs*, we exhibit a rewriting algorithm that allows us to prove that SC-entailment and IC-entailment of BCQs and BUCQs are first-order rewritable, and thus in AC^0 in data complexity (Section 5.2).

Policy	Query language	Confidentiality preservation		Data complexity	
		SC-ent.	IC-ent.	SC-ent.	IC-ent.
Acyclic	BCQ	yes		in AC^0	
Arbitrary	BCQ	yes		coNP-c	
Acyclic	BUCQ	no	yes	in AC^0	in AC^0
Arbitrary	BUCQ	no	yes	coNP-c	coNP-c

Table 1: Results on data complexity and confidentiality preservation. coNP-c stands for coNP-complete.

Our results are summarized in Table 1. An extended version with complete proofs can be found in [Cima *et al.*, 2024].

2 Preliminaries

We employ standard notions of function-free first-order logic (FO), and consider FO formulas using only unary and binary predicates called *concepts* and *roles*, respectively, as in Description Logics, which are fragments of FO suited for conceptual modeling [Baader *et al.*, 2020]. We assume the existence of pairwise-disjoint countably-infinite sets Γ_C, Γ_R, Γ_I, Γ_N, and Γ_V containing *atomic concepts*, *atomic roles*, *constants* (also known as *individuals*), *labeled nulls*, and *variables*, respectively. An FO formula ϕ is sometimes denoted as $\phi(\vec{x})$, where $\vec{x}$ is the sequence of the free variables occurring in ϕ. We also use the term *query* as a synonym of FO formula and *Boolean query* as a synonym of closed FO formula (also called sentence). An FO theory Φ is a set of FO sentences. The semantics of Φ is given in terms of FO interpretations over $\Gamma_C \cup \Gamma_R \cup \Gamma_I$.W.l.o.g., we consider interpretations sharing the same infinite countable domain $\Delta = \Gamma_I$, so that every element in Γ_I is interpreted by itself. In other terms, we use *standard names*, as often customary when one deals with epistemic operators [Calvanese *et al.*, 2007a]. We write $eval(\phi, \mathcal{I})$ to indicate the evaluation of an FO sentence ϕ over an FO interpretation $\mathcal{I}$. A *model* of an FO theory Φ is an FO interpretation satisfying all sentences in Φ. We say that Φ *entails* an FO sentence ϕ, denoted by $\Phi \models \phi$, if $eval(\phi, \mathcal{I})$ is true in every model $\mathcal{I}$ of Φ.

A *Description Logic (DL) ontology* $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$, representing intensional and extensional knowledge, respectively. In this paper, an ABox is a finite set of atoms using predicate symbols from $\Gamma_C \cup \Gamma_R$ and terms from $\Gamma_I \cup \Gamma_N$ (ABoxes of this form are also called quantified ABoxes [Baader *et al.*, 2020]). A model of an ontology $\mathcal{T} \cup \mathcal{A}$ is any model of the FO theory $\mathcal{T} \cup \{\exists \vec{x}\, \phi_\mathcal{A}(\vec{x})\}$,

where $\vec{x}$ is a sequence of variables from Γ_V and $\phi_{\mathcal{A}}(\vec{x})$ is the conjunction of all the atoms of $\mathcal{A}$ in which each labeled null is replaced with a distinct variable $x \in \vec{x}$. With a little abuse of notation, we sometimes treat an ontology $\mathcal{T} \cup \mathcal{A}$ (resp. $\mathcal{A}$) as the FO theory $\mathcal{T} \cup \{\exists \vec{x}\, \phi_{\mathcal{A}}(\vec{x})\}$ (resp. $\{\exists \vec{x}\, \phi_{\mathcal{A}}(\vec{x})\}$). For instance, this allows us to write $\mathcal{T} \cup \mathcal{A} \models \phi$ to intend $\mathcal{T} \cup \{\exists \vec{x}\, \phi_{\mathcal{A}}(\vec{x})\} \models \phi$, where ϕ is an FO sentence.

Our complexity results are given for ontologies expressed in DL-Lite$_{\mathcal{R}}$, a member of the well-known DL-Lite family of DLs [Calvanese *et al.*, 2007b]. A DL-Lite$_{\mathcal{R}}$ TBox $\mathcal{T}$ is a finite set of axioms of the form $B \sqsubseteq B'$ and $R \sqsubseteq R'$ (called concept and role inclusions), or $B \sqsubseteq \neg B'$ and $R \sqsubseteq \neg R'$ (called concept and role disjointnesses), where B and B' (resp., R and R') are predicates of the form A, $\exists S$ or $\exists S^-$ (resp., S or S^-) such that $A \in \Gamma_C$, $S \in \Gamma_R$, S^- is the inverse of S, and the *unqualified existential restrictions* $\exists S$ and $\exists S^-$ represent the set of objects appearing as the first and second argument of S, respectively.

As usual when studying query answering over DL ontologies, we focus on the language of conjunctive queries (and variants thereof). A *conjunctive query* (*CQ*) takes the form of an FO formula $\exists \vec{y} \phi(\vec{x}, \vec{y})$, where $\vec{x} \cup \vec{y} \subseteq \Gamma_V$ and $\phi(\vec{x}, \vec{y})$ is a finite, non-empty conjunction of atoms of the form $\alpha(\vec{t})$, where $\alpha \in \Gamma_C \cup \Gamma_R$, and each term in $\vec{t}$ is either a constant in Γ_I or a variable in $\vec{x} \cup \vec{y}$. We also consider the special CQ $\bot$ and assume that $\mathit{eval}(\bot, \mathcal{I})$ is false for every FO interpretation $\mathcal{I}$. A *union of conjunctive queries* (*UCQ*) is a disjunction $q_1(\vec{x}) \vee \ldots \vee q_n(\vec{x})$ of CQs. For convenience, sometimes we treat UCQs as sets of CQs. Boolean CQs and UCQs, for short, are respectively indicated as *BCQs* and *BUCQs*.

Given a CQ q, we denote by $\mathit{Len}(q)$ the number of atoms in q. Given a UCQ q, $\mathit{MaxLenCQ}(q) = \max_{q' \in q} \mathit{Len}(q')$. We denote by **BCQ** (resp. **BUCQ**) the language of all the BCQs (resp. BUCQs), and, for a positive integer k, by $\mathbf{BCQ}_k$ (resp. $\mathbf{BUCQ}_k$) the language of BCQs (resp. BUCQs) q such that $\mathit{Len}(q) \leq k$ (resp., $\mathit{MaxLenCQ}(q) \leq k$). Given a TBox $\mathcal{T}$, an ABox $\mathcal{A}$, and a Boolean query language $\mathcal{L}_q \subseteq \mathbf{BCQ}$, we denote by $\mathcal{L}_q$-$\mathit{Cons}(\mathcal{T} \cup \mathcal{A})$ the set of Boolean queries $q \in \mathcal{L}_q$ that are logical consequences of $\mathcal{T} \cup \mathcal{A}$. Formally: $\mathcal{L}_q$-$\mathit{Cons}(\mathcal{T} \cup \mathcal{A}) = \{q \in \mathcal{L}_q \mid \mathcal{T} \cup \mathcal{A} \models q\}$

A *ground substitution* for a sequence $\vec{x} = x_1, \ldots, x_k$ of variables is a sequence of constants $\vec{c} = c_1, \ldots, c_k$. Furthermore, if $\vec{x}$ are the free variables of a FO formula $\phi(\vec{x})$, we indicate as $\phi(\vec{c})$ the FO sentence obtained from $\phi(\vec{x})$ by replacing each x_i with c_i, for $1 \leq i \leq k$.

We recall that query answering of UCQs in DL-Lite$_{\mathcal{R}}$ is first-order rewritable, i.e. for every DL-Lite$_{\mathcal{R}}$ TBox $\mathcal{T}$ and UCQ $q(\vec{x})$, it is possible to compute an FO query $q_r(\vec{x})$ such that, for every ground substitution $\vec{c}$ of $\vec{x}$, $\mathcal{T} \cup \mathcal{A} \models q(\vec{c})$ iff $\mathcal{A} \models q_r(\vec{c})$. To compute $q_r(\vec{x})$, we use the algorithm *PerfectRef* presented in [Calvanese *et al.*, 2007b], for which the following property holds.

Proposition 1. *Let $\mathcal{T}$ be a DL-Lite$_{\mathcal{R}}$ TBox and let $q(\vec{x})$ be a UCQ. For every ABox $\mathcal{A}$ and every ground substitution $\vec{c}$ for $\vec{x}$, we have that $\mathcal{T} \cup \mathcal{A} \models q(\vec{c})$ if and only if $\mathcal{A} \models q_r(\vec{c})$, where $q_r(\vec{x}) = \mathit{PerfectRef}(q(\vec{x}), \mathcal{T})$.*

We point out that, by construction, $q_r = \mathit{PerfectRef}(q, \mathcal{T})$ is a UCQ and that $\mathit{MaxLenCQ}(q_r) = \mathit{MaxLenCQ}(q)$.

3 Framework

In this section, we describe our CQE framework. We first give the notion of epistemic dependencies that we use in the policies, then present the notion of censor, and finally provide two notions of query entailment in our novel framework.

Epistemic dependencies. The policy $\mathcal{P}$ of our framework is a finite set of *epistemic dependencies*, each of which can be seen as a *domain-independent EQL-Lite(CQ)* [Calvanese *et al.*, 2007a] sentence defined as follows.

Definition 1. *An* epistemic dependency (ED) *is a sentence δ of the following form:*

$$\forall \vec{x}_1, \vec{x}_2 (K q_b(\vec{x}_1, \vec{x}_2) \rightarrow K q_h(\vec{x}_2)) \tag{1}$$

where $q_b(\vec{x}_1, \vec{x}_2)$ is a CQ with free variables $\vec{x}_1 \cup \vec{x}_2$, $q_h(\vec{x}_2)$ is a CQ with free variables $\vec{x}_2$, and K is a modal operator. The variables $\vec{x}_2$ are called the frontier variables of δ.

Intuitively, an ED of form (1) should be read as follows: if the sentence $q_b(\vec{c}_1, \vec{c}_2)$ is *known* to hold, then the sentence $q_h(\vec{c}_2)$ is *known* to hold, for any ground substitutions $\vec{c}_1$ and $\vec{c}_2$ for $\vec{x}_1$ and $\vec{x}_2$, respectively. More formally, we define when an FO theory Φ *satisfies* an ED δ, denoted $\Phi \models_{\mathsf{EQL}} \delta$. To this aim, we consider the set E of all FO models of Φ, and say that $\Phi \models_{\mathsf{EQL}} \delta$ if, for every ground substitutions $\vec{c}_1$ for $\vec{x}_1$ and $\vec{c}_2$ for $\vec{x}_2$, the fact that $\mathit{eval}(q_b(\vec{c}_1, \vec{c}_2), \mathcal{I})$ is true for every $\mathcal{I} \in E$ implies that $\mathit{eval}(q_h(\vec{c}_2), \mathcal{I})$ is true for every $\mathcal{I} \in E$.

We say that Φ *satisfies* a policy $\mathcal{P}$ (denoted $\Phi \models_{\mathsf{EQL}} \mathcal{P}$) if Φ satisfies δ, for each $\delta \in \mathcal{P}$. We remark that, as already said, EDs of the form (1) have been originally introduced in [Console and Lenzerini, 2020], although in a slightly more general form, to express integrity constraints in ontology-based data management.Then, the notion of ED satisfaction defined above is essentially as in [Console and Lenzerini, 2020].

Example 1. *Suppose that company C_A wants to share certain user-profiling data with a company C_B for targeted advertising. This is not allowed in general, but only in some countries with a special regulation that enables sharing based on the users' consent. C_A may use the following ED in the policy to enable C_B to access only data compliant with the above requirements:*

$$\delta_4 = \forall x, y \big(K\, \mathsf{profiledActivity}(x, y) \rightarrow K\, \exists z (\mathsf{citOf}(x, z) \wedge \mathsf{SR}(z) \wedge \mathsf{Consent}(x)) \big)$$

In the rule, profiledActivity *associates a user with her profiling-data,* citOf *relates a user to the country of which she is a citizen,* SR *denotes countries with special regulation, and* Consent *denotes users who have given their consent.*

Suppose that C_A also wants C_B not to be able to associate a user with her real identity, and that this is possible by collecting the person's name and her date of birth at the same time. To this aim, C_A also specifies the following ED:

$$\delta_5 = \forall x, y, z \big(K(\mathsf{name}(x, y) \wedge \mathsf{dateB}(x, z)) \rightarrow K \bot \big) \quad \blacksquare$$

CQ-censors. As already said, censors are used to enforce confidentiality on an ontology coupled with a data protection policy. Among various definitions of censors proposed in the literature, we adapt here the one investigated in [Lembo *et al.*, 2019] to properly deal with policies constituted by sets of EDs. To this aim, it is convenient to first define CQE instances in our novel epistemic CQE framework.

Definition 2 (CQE instance)**.** *An $\mathcal{L}_\mathcal{T}$ CQE instance is a triple $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where $\mathcal{T}$ is a TBox in the DL $\mathcal{L}_\mathcal{T}$, $\mathcal{A}$ is an ABox such that $\mathcal{T} \cup \mathcal{A}$ has at least one model, and $\mathcal{P}$ is a policy (i.e., a finite set of EDs) such that $\mathcal{T} \models_{\mathsf{EQL}} \mathcal{P}$.*

Hereinafter, if the DL $\mathcal{L}_\mathcal{T}$ is not specified, we implicitly intend any possible DL.

The notion of (optimal) CQ-censors is then as follows.

Definition 3 ((optimal) CQ-censor)**.** *A* CQ-censor *of a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ is a subset $\mathcal{C}$ of **BCQ**-$Cons(\mathcal{T} \cup \mathcal{A})$ such that $\mathcal{T} \cup \mathcal{C} \models_{\mathsf{EQL}} \mathcal{P}$.*

An optimal CQ-censor *of a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ is a CQ-censor of $\mathcal{E}$ such that there exists no CQ-censor $\mathcal{C}'$ of $\mathcal{E}$ such that $\mathcal{C}' \supset \mathcal{C}$. We denote by OptCQCens($\mathcal{E}$) the set of optimal CQ-censors of $\mathcal{E}$.*

Entailment. As in [Lembo *et al.*, 2019], in this paper CQE amounts to reason over all the possible optimal censors, according to the following definition of SC-entailment.

Definition 4 (SC-entailment)**.** *A CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$* Skeptically-entails a BUCQ q under CQ-Censors *(in short,* SC-entails *q), denoted by $\mathcal{E} \models_{SC} q$, if $\mathcal{T} \cup \mathcal{C} \models q$ for every $\mathcal{C} \in$ OptCQCens($\mathcal{E}$).*

We also consider the following sound approximation of SC-entailment.

Definition 5 (IC-entailment)**.** *A CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$* entails a BUCQ q under the Intersection of CQ-Censors *(in short,* IC-entails *q), denoted by $\mathcal{E} \models_{IC} q$, if $\mathcal{T} \cup \mathcal{C}_{int}(\mathcal{E}) \models q$, where $\mathcal{C}_{int}(\mathcal{E}) = \bigcap_{\mathcal{C} \in OptCQCens(\mathcal{E})} \mathcal{C}$.*

Example 2. *Consider the CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where $\mathcal{T} = \emptyset$, $\mathcal{P} = \{\delta_4, \delta_5\}$, with δ_4 and δ_5 being the EDs illustrated in Example 1, and the ABox $\mathcal{A}$ is as follows:*

$$\mathcal{A} = \{\mathsf{profiledActivity}(\mathsf{p}_1, \mathsf{act}_1), \mathsf{Consent}(\mathsf{p}_1), \mathsf{citOf}(\mathsf{p}_1, n_1), \mathsf{SR}(n_1), \mathsf{name}(\mathsf{p}_1, \mathsf{ann}), \mathsf{dateB}(\mathsf{p}_1, \mathsf{date}_1), \mathsf{profiledActivity}(\mathsf{p}_2, \mathsf{act}_2), \mathsf{citOf}(\mathsf{p}_2, n_1)\},$$

where $n_1 \in \Gamma_N$ while all the other terms used in $\mathcal{A}$ are constants in Γ_I. Now, consider the following four BCQs:

$q_1 = \exists y\, (\mathsf{profiledActivity}(\mathsf{p}_1, \mathsf{act}_1) \wedge \mathsf{citOf}(\mathsf{p}_1, y) \wedge \mathsf{SR}(y))$

$q_2 = \mathsf{profiledActivity}(\mathsf{p}_2, \mathsf{act}_2)$

$q_3 = \exists y\, \mathsf{profiledActivity}(y, \mathsf{act}_2)$

$q_4 = \mathsf{profiledActivity}(\mathsf{p}_1, \mathsf{act}_1) \wedge \mathsf{name}(\mathsf{p}_1, \mathsf{ann})$

For $X \in \{SC, IC\}$, one can verify that $\mathcal{E} \models_X q_1$ because p_1 gave the consent and she is a citizen of some country (n_1) with special regulation. Conversely, one can see that $\mathcal{E} \not\models_X q_2$ because p_2 did not give the consent. Nevertheless, it is easy to verify that $q_3 \in \mathcal{C}$ for each optimal CQ-censor $\mathcal{C}$ of $\mathcal{E}$, and therefore $\mathcal{E} \models_X q_3$. Finally, we have that $\mathcal{E} \not\models_X q_4$ because there exists an optimal CQ-censor $\mathcal{C}$ of $\mathcal{E}$ such that $\mathsf{dateB}(\mathsf{p}_1, \mathsf{date}_1) \in \mathcal{C}$, thus implying that $\mathsf{name}(\mathsf{p}_1, \mathsf{ann}) \notin \mathcal{C}$ (otherwise δ_5 would be violated). ∎

In the above example, note that SC- and IC-entailment coincide for all queries. As shown in the subsequent result, this always holds in the case of entailment of BCQs.

Theorem 1. *For every CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ and for every BCQ q, we have that $\mathcal{E} \models_{SC} q$ iff $\mathcal{E} \models_{IC} q$.*

Proof. First, it is easy to verify that, for every $\mathcal{C}$ that is an optimal CQ-censor of $\mathcal{E}$, and for every BCQ q, $\mathcal{T} \cup \mathcal{C} \models q$ iff $q \in \mathcal{C}$; moreover, $\mathcal{T} \cup \mathcal{C}_{int}(\mathcal{E}) \models q$ iff $q \in \mathcal{C}_{int}(\mathcal{E})$. Now, let q be a BCQ. If $\mathcal{E} \models_{\text{SC}} q$, then q belongs to all the optimal CQ-censors of $\mathcal{E}$, and thus $q \in \mathcal{C}_{int}(\mathcal{E})$, which implies that $\mathcal{E} \models_{\text{IC}} q$. Conversely, if $\mathcal{E} \not\models_{\text{SC}} q$, then there exists an optimal CQ-censor of $\mathcal{E}$ that does not contain q, hence $q \notin \mathcal{C}_{int}(\mathcal{E})$, which implies that $\mathcal{E} \not\models_{\text{IC}} q$. □

On the other hand, the next example shows that the same result does not hold for entailment of BUCQs.

Example 3. *Recall Example 2, and consider the BUCQ $q = \mathsf{name}(\mathsf{p}_1, \mathsf{ann}) \vee \mathsf{dateB}(\mathsf{p}_1, \mathsf{date}_1)$. While we have that $\mathcal{E} \models_{SC} q$, because either $\mathsf{name}(\mathsf{p}_1, \mathsf{ann}) \in \mathcal{C}$ or $\mathsf{dateB}(\mathsf{p}_1, \mathsf{date}_1) \in \mathcal{C}$ holds for every $\mathcal{C} \in$ OptCQCens($\mathcal{E}$), it is easy to see that $\mathcal{E} \not\models_{IC} q$ as neither $\mathsf{name}(\mathsf{p}_1, \mathsf{ann})$ nor $\mathsf{dateB}(\mathsf{p}_1, \mathsf{date}_1)$ belong to $\mathcal{C}_{int}(\mathcal{E}) = \bigcap_{\mathcal{C} \in OptCQCens(\mathcal{E})} \mathcal{C}$.* ∎

4 Confidentiality Preservation

In this section, we investigate the notion of confidentiality used in this paper. We adopt a similar approach to the one described in [Biskup and Weibert, 2008] for relational databases. Intuitively, under such an approach an entailment semantics preserves confidentiality if, for every CQE instance $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ and every finite set $\mathcal{Q}$ of queries, the answers to such queries are the same as if they were obtained from another CQE instance $\langle \mathcal{T}, \mathcal{A}', \mathcal{P} \rangle$ such that $\mathcal{T} \cup \mathcal{A}' \models_{\mathsf{EQL}} \mathcal{P}$.

We now describe this property formally. First, for a TBox $\mathcal{T}$, a policy $\mathcal{P}$, two ABoxes $\mathcal{A}$ and $\mathcal{A}'$, a set $\mathcal{Q}$ of BUCQs, and $X \in \{\text{SC}, \text{IC}\}$, we say that $\mathcal{A}$ and $\mathcal{A}'$ are *$\mathcal{Q}$-indistinguishable for X-entailment with respect to* $(\mathcal{T}, \mathcal{P})$ if, for every $q \in \mathcal{Q}$, we have that $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \models_X q$ iff $\langle \mathcal{T}, \mathcal{A}', \mathcal{P} \rangle \models_X q$.

Definition 6. *Given a query language $\mathcal{L}_q \subseteq$ **BUCQ** and a DL $\mathcal{L}_\mathcal{T}$, for $X \in \{SC, IC\}$, we say that X-*entailment preserves confidentiality for $\mathcal{L}_q$ in $\mathcal{L}_\mathcal{T}$ *if, for every $\mathcal{L}_\mathcal{T}$ CQE instance $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, and for every finite set $\mathcal{Q} \subseteq \mathcal{L}_q$, there exists an ABox $\mathcal{A}'$ such that (i) $\mathcal{T} \cup \mathcal{A}' \models_{\mathsf{EQL}} \mathcal{P}$ and (ii) $\mathcal{A}$ and $\mathcal{A}'$ are $\mathcal{Q}$-indistinguishable for X-entailment w.r.t. $(\mathcal{T}, \mathcal{P})$.*

For both SC-entailment and IC-entailment, we now investigate confidentiality-preservation for **BUCQ** in DL-Lite$_\mathcal{R}$.

Hereinafter, with a slight abuse of notation, given a policy $\mathcal{P}$, we denote by *MaxLenCQ*($\mathcal{P}$) the maximum length (number of atoms) of a CQ occurring within the scope of the K operator in $\mathcal{P}$.

Proposition 2. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance. For every integer $k > 0$, there exists a DL-Lite$_\mathcal{R}$ CQE instance $\mathcal{E}' = \langle \mathcal{T}, \mathcal{A}', \mathcal{P} \rangle$ such that (i) $\mathcal{T} \cup \mathcal{A}' \models_{\mathsf{EQL}} \mathcal{P}$ and (ii) for every $q \in$ **BCQ**$_k$, $\mathcal{E} \models_{SC} q$ iff $\mathcal{E}' \models_{SC} q$.*

Proof (sketch). Let $\mathcal{A}'$ be the ABox isomorphic to the finite set of BCQs $\{q \in \mathbf{BCQ}_h \mid \mathcal{E} \models_{\text{SC}} q\}$, where $h = \max(k, \mathit{MaxLenCQ}(\mathcal{P}))$. It is easy to verify that $\mathcal{T} \cup \mathcal{A}' \models_{\mathsf{EQL}} \mathcal{P}$, from which it follows that $\mathcal{E} \models_{\text{SC}} q$ iff $\mathcal{E}' \models_{\text{SC}} q$ for every $q \in \mathbf{BCQ}_k$. □

With the above property at hand, we can prove that SC-entailment preserves confidentiality for **BCQ** in DL-Lite$_\mathcal{R}$. We show, however, that the same does not hold for **BUCQ**.

Theorem 2. *SC-entailment preserves confidentiality for **BCQ** in DL-Lite$_\mathcal{R}$, whereas it does not preserve confidentiality for **BUCQ** in DL-Lite$_\mathcal{R}$.*

Proof. The first statement is an easy consequence of Proposition 2, when we assume that k is the maximum length of a BCQ in $\mathcal{Q}$. For the second statement, we give a counterexample. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where $\mathcal{T} = \emptyset$, $\mathcal{A} = \{C_1(o), C_2(o)\}$ and $\mathcal{P} = \{\forall x(K(C_1(x) \wedge C_2(x)) \rightarrow K\bot)\}$. Consider also the BUCQ $q = C_1(o) \vee C_2(o)$. It is easy to see that no ABox $\mathcal{A}'$ is such that $\mathcal{T} \cup \mathcal{A}' \models_{\mathsf{EQL}} \mathcal{P}$, and $\mathcal{A}$ and $\mathcal{A}'$ are $\{q\}$-indistinguishable for SC-entailment w.r.t. $(\mathcal{T}, \mathcal{P})$. □

The above proof also shows that SC-entailment does not preserve confidentiality for **BUCQ** in DL-Lite$_\mathcal{R}$ even when EDs are restricted to be acyclic (see Definition 8).

However, it turns out that confidentiality in the case of BUCQs in DL-Lite$_\mathcal{R}$ is preserved by IC-entailment.

Proposition 3. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance. For every integer $k > 0$, there exists a DL-Lite$_\mathcal{R}$ CQE instance $\mathcal{E}' = \langle \mathcal{T}, \mathcal{A}', \mathcal{P} \rangle$ such that (i) $\mathcal{T} \cup \mathcal{A}' \models_{\mathsf{EQL}} \mathcal{P}$ and (ii) for every $q \in \boldsymbol{BCQ}_k$, $\mathcal{E} \models_{IC} q$ iff $\mathcal{E}' \models_{IC} q$.*

From the above property, we get the following result.

Theorem 3. *IC-entailment preserves confidentiality for **BUCQ** in DL-Lite$_\mathcal{R}$.*

5 Algorithms and Complexity Results

In this section, we analyze the data complexity of SC- and IC-entailment of B(U)CQs for DL-Lite$_\mathcal{R}$ CQE instances.

We study the decision problems associated with the query answering problem under SC- and IC-entailment. Specifically, we consider the following recognition problem X-REC$[\mathcal{L}_q]$, which is parametric w.r.t. a Boolean query language $\mathcal{L}_q$ and $X \in \{\text{SC}, \text{IC}\}$:

Input: A DL-Lite$_\mathcal{R}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, a Boolean query $q \in \mathcal{L}_q$
Question: Does $\mathcal{E} \models_X q$?

We are interested in the data complexity [Vardi, 1982] version of the above problem, which is the complexity where only the ABox $\mathcal{A}$ is regarded as the input while all the other components are assumed to be fixed.

5.1 Arbitrary Policies

We start by analyzing the complexity of SC-entailment of BCQs and BUCQs.

Lemma 1. *SC-REC[**BCQ**] is coNP-hard in data complexity.*

Proof (sketch). We show a reduction of 3-CNF, a well-known NP-hard problem, to the complement of SC-REC[**BCQ**]. Let $\mathcal{T}$ be the empty TBox, and let $\mathcal{P}$ contain the following EDs:

$$\forall x, y, v, z \left(K(C_1(x, y) \wedge V_1(x, v) \wedge V(y, v) \wedge N(x, z) \wedge S(x)) \rightarrow KS(z)\right)$$
$$\forall x, y, v, z \left(K(C_2(x, y) \wedge V_2(x, v) \wedge V(y, v) \wedge N(x, z) \wedge S(x)) \rightarrow KS(z)\right)$$
$$\forall x, y, v, z \left(K(C_3(x, y) \wedge V_3(x, v) \wedge V(y, v) \wedge N(x, z) \wedge S(x)) \rightarrow KS(z)\right)$$
$$\forall x \left(K(V(x, f) \wedge V(x, t)) \rightarrow K\bot\right)$$

Given a 3-CNF formula ϕ with m clauses, we represent every clause of ϕ in the ABox $\mathcal{A}$ through the roles $C_1, C_2, C_3, V_1, V_2, V_3$: e.g., if the i-th clause of ϕ is $\neg a \vee b \vee \neg c$, we add to $\mathcal{A}$ the assertions $C_1(i, a), C_2(i, b), C_3(i, c), V_1(i, f), V_2(i, t), V_3(i, f)$. Moreover, $\mathcal{A}$ contains the assertions

$$\{V(a, f), V(a, t) \mid a \in \mathit{PV}(\phi)\} \cup \{S(i), N(i, i+1) \mid 1 \leq i \leq m\}$$

where $\mathit{PV}(\phi)$ are the propositional variables of ϕ. It can be verified that ϕ is satisfiable iff $\langle \emptyset, \mathcal{A}, \mathcal{P} \rangle \not\models_{\text{SC}} S(1)$. □

In order to prove a matching coNP upper bound for SC-REC[**BUCQ**], we introduce a notion of consequences of a set of BCQs $\mathcal{C}$ with respect to a TBox $\mathcal{T}$ and a policy $\mathcal{P}$.

Definition 7. *Let $\mathcal{T}$ be a DL-Lite$_\mathcal{R}$ TBox, $\mathcal{P}$ a policy and $\mathcal{C} \subseteq$ **BCQ**. We define PolicyCons$(\mathcal{T}, \mathcal{P}, \mathcal{C})$ as the smallest set $\mathcal{S} \subseteq$ **BCQ** such that:*

(i) $\mathcal{C} \subseteq \mathcal{S}$;

(ii) for every ED $\forall \vec{x}_1, \vec{x}_2(Kq_b(\vec{x}_1, \vec{x}_2) \rightarrow Kq_h(\vec{x}_2))$ in $\mathcal{P}$ and ground substitutions $\vec{t}_1$ and $\vec{t}_2$ for $\vec{x}_1$ and $\vec{x}_2$, respectively, if $\mathcal{T} \cup \mathcal{S} \models q_b(\vec{t}_1, \vec{t}_2)$ then $q_h(\vec{t}_2) \in \mathcal{S}$.

It can be immediately verified that *PolicyCons*$(\mathcal{T}, \mathcal{P}, \mathcal{C})$ can be computed in polynomial time w.r.t. the size of $\mathcal{C}$.

The following property can be easily derived from the previous definition and the definition of optimal CQ-censor.

Lemma 2. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance. For every $\mathcal{C} \subseteq$ **BCQ**, there exists an optimal CQ-censor of $\mathcal{E}$ containing $\mathcal{C}$ iff PolicyCons$(\mathcal{T}, \mathcal{P}, \mathcal{C}) \subseteq$ **BCQ**-$Cons(\mathcal{T} \cup \mathcal{A})$.*

We are now ready to provide an algorithm for checking SC-entailment of BUCQs.

Algorithm 1: SC-Entails$(\mathcal{E}, q)$

input: DL-Lite$_\mathcal{R}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, BUCQ q
$k \leftarrow \max(\mathit{MaxLenCQ}(q), \mathit{MaxLenCQ}(\mathcal{P}))$;
if there exists $\mathcal{C} \subseteq \boldsymbol{BCQ}_k$-$Cons(\mathcal{T} \cup \mathcal{A})$ such that:
  (i) $\mathcal{T} \cup \mathcal{C} \models_{\mathsf{EQL}} \mathcal{P}$ and
  (ii) $\mathcal{T} \cup \mathcal{C} \not\models q$ and
  (iii) for every $q' \in \boldsymbol{BCQ}_k$-$Cons(\mathcal{T} \cup \mathcal{A}) \setminus \mathcal{C}$, *PolicyCons*$(\mathcal{T}, \mathcal{P}, \mathcal{C} \cup \{q'\}) \not\subseteq \boldsymbol{BCQ}_k$-$Cons(\mathcal{T} \cup \mathcal{A})$
  then return $false$;
return $true$;

Example 4. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where $\mathcal{T} = \{A \sqsubseteq D\}$, $\mathcal{P} = \{\forall x(K(D(x) \wedge C(x)) \rightarrow K\bot), \forall x(KB(x) \rightarrow KA(x))\}$, and $\mathcal{A} = \{A(o), B(o), C(o)\}$.*

Let now $\mathcal{C}$ be a maximal subset of $\boldsymbol{BCQ}_2$-$Cons(\mathcal{T} \cup \mathcal{A})$ such that $\mathcal{T} \cup \mathcal{C} \not\models A(o) \vee B(o) \vee D(o)$. One can see that, for the BCQ $q = B(o)$, SC-Entails$(\mathcal{E}, q)$ returns false as $\mathcal{C}$ satisfies conditions (i), (ii), and (iii) of the algorithm. In particular, PolicyCons$(\mathcal{T}, \mathcal{P}, \mathcal{C} \cup \{q\})$ contains $B(o)$, $A(o)$, $D(o)$, and $\bot$, hence it is not a subset of $\boldsymbol{BCQ}_2$-$Cons(\mathcal{T} \cup \mathcal{A})$. ■

Lemma 3. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance, and let q be a BUCQ. The algorithm SC-Entails$(\mathcal{E}, q)$ returns true iff $\mathcal{E} \models_{SC} q$.*

Proof (sketch). First, using Lemma 2, it is easy to derive that a set of BCQs $\mathcal{C}$ satisfying conditions (i), (ii), and (iii) exists iff there exists an optimal CQ-censor of $\mathcal{E}$ containing $\mathcal{C}$ and not containing q (and hence iff $\mathcal{E} \not\models_{\text{SC}} q$). Then, it can be shown that, for a DL-Lite$_\mathcal{R}$ TBox $\mathcal{T}$ and a set of BCQs $\mathcal{C}$ that is closed under subqueries (i.e. for every BCQ $q \in \mathcal{C}$, $\mathcal{C}$ contains all the subqueries of q), $\mathcal{T} \cup \mathcal{C} \models q$ iff $\mathcal{T} \cup \mathcal{C}_h \models q$, where $h = \mathit{MaxLenCQ}(q)$ and $\mathcal{C}_h = \mathcal{C} \cap \mathbf{BCQ}_k$. This is the key property that allows us to prove that, if a set of BCQs $\mathcal{C}$ satisfying conditions (i), (ii), and (iii) exists, then there exists a set of BCQs $\mathcal{C} \subset \boldsymbol{BCQ}_k\text{-}Cons(\mathcal{T} \cup \mathcal{A})$ satisfying such conditions, which implies the correctness of the algorithm. □

The next theorem follows from Lemma 1, Lemma 3, and from the fact that the previous algorithm can be executed in nondeterministic polynomial time in data complexity.

Theorem 4. *SC-REC[**BUCQ**] is coNP-complete in data complexity.*

We now give a second algorithm, which makes use of the above algorithm SC-Entails to check IC-entailment.

Algorithm 2: IC-Entails($\mathcal{E}, q$)

input: DL-Lite$_\mathcal{R}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, BUCQ q

foreach BCQ $q' \in q$ **do**
  if SC-Entails($\mathcal{E}, q'$) **then return** $true$;
return $false$;

Lemma 4. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance and let q be a BUCQ. The algorithm IC-Entails($\mathcal{E}, q$) returns true iff $\mathcal{E} \models_{IC} q$.*

Proof. First, from Definition 5, if $\mathcal{T}$ is a DL-Lite$_\mathcal{R}$ TBox, then $\mathcal{E} \models_{\text{IC}} q$ iff there exists a BCQ $q' \in q$ such that $\mathcal{E} \models_{\text{IC}} q'$. Then, by Theorem 1, $\mathcal{E} \models_{\text{IC}} q'$ iff $\mathcal{E} \models_{\text{SC}} q'$. Therefore, by Lemma 3 the thesis follows. □

From Lemma 1, Lemma 4, Theorem 1, and from the fact that the algorithm SC-Entails($\mathcal{E}, q$) can be executed in nondeterministic polynomial time in data complexity, we obtain:

Theorem 5. *IC-REC[**BUCQ**] is coNP-complete in data complexity.*

5.2 Acyclic Policies

Given the intractability results for the set of all DL-Lite$_\mathcal{R}$ CQE instances presented above, in this section we focus on a subclass of DL-Lite$_\mathcal{R}$ CQE instances, whose policies enjoy an acyclicity property.

First, we extend the notion of first-order rewritability defined over ground ABoxes to the case of ABoxes with labeled nulls and to the problems of SC-entailment and IC-entailment of BUCQs. Given an ABox $\mathcal{A}$, we define the FO interpretation $\mathcal{I}_\mathcal{A}$ over the predicates $\Gamma_C \cup \Gamma_R$ plus the additional new concept *Ind*, and the constants $\Gamma_I \cup \Gamma_N$ (i.e. in $\mathcal{I}_\mathcal{A}$ we consider the symbols from Γ_N as ordinary constants):

- $\Delta^{\mathcal{I}_\mathcal{A}} = \Gamma_I \cup \Gamma_N$;
- $a^{\mathcal{I}_\mathcal{A}} = a$ for every $a \in \Gamma_I \cup \Gamma_N$;
- for every concept name C, $C^{\mathcal{I}_\mathcal{A}} = \{a \mid C(a) \in \mathcal{A}\}$;
- for every role name R, $R^{\mathcal{I}_\mathcal{A}} = \{(a, b) \mid R(a, b) \in \mathcal{A}\}$;
- $\mathit{Ind}^{\mathcal{I}_\mathcal{A}} = \Gamma_I$.

Given a TBox $\mathcal{T}$, a policy $\mathcal{P}$ and a BUCQ q, and $X \in \{\text{SC}, \text{IC}\}$, we say that a FO sentence q' is a *first-order rewriting of X-entailment of q for $\mathcal{T}$ and $\mathcal{P}$* if, for every ABox $\mathcal{A}$, $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \models_X q$ iff $\mathit{eval}(q', \mathcal{I}_\mathcal{A})$ is true.

Our goal now is to define an algorithm that, for a DL-Lite$_\mathcal{R}$ TBox $\mathcal{T}$ and a policy $\mathcal{P}$, is able to construct a first-order rewriting of SC-entailment of q for $\mathcal{T}$ and $\mathcal{P}$. This is not possible in general, given the coNP-completeness result provided by Theorem 5. Therefore, we now define the subclass of policies that are acyclic for a DL-Lite$_\mathcal{R}$ TBox.

Given a DL-Lite$_\mathcal{R}$ TBox $\mathcal{T}$ and a policy $\mathcal{P}$, the *dependency graph* of $\mathcal{T}$ and $\mathcal{P}$, denoted by $G(\mathcal{T}, \mathcal{P})$, is the directed graph defined as follows: (i) the set of nodes of $G(\mathcal{T}, \mathcal{P})$ is the set of predicates occurring in $\mathcal{T} \cup \mathcal{P}$; (ii) there is a P-edge from node p_1 to node p_2 in $G(\mathcal{T}, \mathcal{P})$ if and only if there exists an epistemic dependency of the form (1) in $\mathcal{P}$ such that p_1 occurs in q_b and p_2 occurs in q_h; (iii) there is a T-edge from node p_1 to node p_2 in $G(\mathcal{T}, \mathcal{P})$ if and only if there is a concept or role inclusion in $\mathcal{T}$ such that p_1 occurs in the left-hand side and p_2 occurs in the right-hand side of the inclusion.

Definition 8. *Given a DL-Lite$_\mathcal{R}$ TBox $\mathcal{T}$ and a policy $\mathcal{P}$, we say that $\mathcal{P}$ is* acyclic for $\mathcal{T}$ *if there exists no cycle in $G(\mathcal{T}, \mathcal{P})$ involving a P-edge.*

Informally, the graph $G(\mathcal{T}, \mathcal{P})$ represents the logical dependencies between the predicates in $\mathcal{T}$ and $\mathcal{P}$: a P-edge (resp., a T-edge) from p_1 to p_2 means that predicate p_1 may have a direct implication on p_2 through $\mathcal{P}$ (resp., through $\mathcal{T}$). The notion of acyclicity defined above guarantees that, if (p_1, p_2) is a P-edge in $G(\mathcal{T}, \mathcal{P})$, then there is no path from p_2 to p_1, i.e. p_2 does not have any (direct or indirect) implication on p_1.

Example 5. *The following ED aims to hide the hierarchical structure of an organization unless it is public.*

$$\delta_6 = \forall x, y(K\mathsf{hasPosition}(x, y) \rightarrow K\,\exists z(\mathsf{worksIn}(x, z) \wedge \mathsf{PublicOffice}(z))))$$

*Moreover, we want to hide the fact that a person collaborates with a secret service unless she holds an important position (*KeyPosition*), for example, she is the director. The following ED achieves this goal:*

$$\delta_7 = \forall x(K\,\exists y(\mathsf{collaborate}(x, y) \wedge \mathsf{SecService}(y)) \rightarrow K\,\exists z(\mathsf{hasPosition}(x, z) \wedge \mathsf{KeyPosition}(z)))$$

Let the policy $\mathcal{P}$ be $\mathcal{P} = \{\delta_6, \delta_7\}$. For the empty TBox $\mathcal{T} = \emptyset$, one can see that $\mathcal{P}$ is acyclic for $\mathcal{T}$. Conversely, for the DL-Lite$_\mathcal{R}$ TBox $\mathcal{T} = \{\mathsf{worksIn} \sqsubseteq \mathsf{collaborate}\}$, $\mathcal{P}$ is not acyclic for $\mathcal{T}$, since there is a cycle in $G(\mathcal{T}, \mathcal{P})$ constituted by the P-edges (collaborate, hasPosition) *and* (hasPosition, worksIn) *and the T-edge* (worksIn, collaborate). ■

With this notion in place, we can now describe the decision problem we are going to study. Specifically, for $X \in \{\text{SC}, \text{IC}\}$, we consider the recognition problem X-AREC[$\mathcal{L}_q$], which is parametric w.r.t. a Boolean query language $\mathcal{L}_q$. X-AREC[$\mathcal{L}_q$] is defined exactly as X-REC[$\mathcal{L}_q$]

except that the input DL-Lite$_\mathcal{R}$ CQE instances $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ are such that the policy $\mathcal{P}$ is acyclic for $\mathcal{T}$.

From now on, given a set of CQs $\mathcal{Q}$,[1] we denote by *And*($\mathcal{Q}$) the CQ $\exists \vec{y}(\bigwedge_{\exists \vec{z} \phi \in \mathcal{Q}} \phi)$, where $\vec{y}$ is a sequence of all the existentially quantified variables occurring in $\mathcal{Q}$.

Given a TBox $\mathcal{T}$, a policy $\mathcal{P}$ and a CQ $q(\vec{x})$, we denote by $\phi_{pc}(\mathcal{T}, \mathcal{P}, q(\vec{x}))$ the CQ (with free variables $\vec{x}$) *And*(*PolicyCons*($\mathcal{T}, \mathcal{P}, \{q(\vec{x})\}$)), where *PolicyCons*($\mathcal{T}, \mathcal{P}, \{q(\vec{x})\}$) is as in Definition 7, considering the free variables in $\vec{x}$ as new constant symbols.

Lemma 5. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance, let $\mathcal{P}$ be an acyclic policy for $\mathcal{T}$, and let $q(\vec{x})$ be a CQ. For every ground substitution $\vec{c}$ for $\vec{x}$, there exists an optimal CQ-censor of $\mathcal{E}$ that contains the BCQ $q(\vec{c})$ if and only if eval($q_r(\vec{c}), \mathcal{I}_\mathcal{A}$) is true, where $q_r(\vec{x})$ = PerfectRef($\phi_{pc}(\mathcal{T}, \mathcal{P}, q(\vec{x})), \mathcal{T}$).*

Given a BCQ q, we say that a BCQ q' is a *clash for q in $\mathcal{E}$* if there exists an optimal CQ-censor of $\mathcal{E}$ containing q' and there exists no optimal CQ-censor of $\mathcal{E}$ containing both q and q'. Given a BUCQ q, we say that a BCQ q' is a clash for q in $\mathcal{E}$ if for every $q'' \in q$, q' is a clash for q'' in $\mathcal{E}$.

Now, let q be a BUCQ and let $q'(\vec{x})$ be a CQ. We denote by *Clash*($q, q'(\vec{x}), \mathcal{T}, \mathcal{P}$) the following FO formula (with free variables $\vec{x}$):

$$\textit{PerfectRef}(\phi_{pc}(\mathcal{T}, \mathcal{P}, q'(\vec{x})), \mathcal{T}) \wedge \Big(\bigwedge_{q_i \in q} \neg \textit{PerfectRef}(\phi_{pc}(\mathcal{T}, \mathcal{P}, \textit{And}(\{q'(\vec{x}), q_i)\}), \mathcal{T}) \Big)$$

Using Lemma 5, we are able to prove the following property.

Lemma 6. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance, let $q \in$ **BUCQ**, let $q'(\vec{x})$ be a CQ, and let $q_i \in$ **BCQ**-$Cons(\mathcal{T} \cup \mathcal{A})$ for every $q_i \in q$. Then, for every ground substitution $\vec{c}$ for $\vec{x}$, $q'(\vec{c})$ is a clash for q in $\mathcal{E}$ iff eval($q_{cl}(\vec{c}), \mathcal{I}_\mathcal{A}$) is true, where $q_{cl}(\vec{x})$ = Clash($q, q'(\vec{x}), \mathcal{T}, \mathcal{P}$).*

It is now possible to show that, in the case of DL-Lite$_\mathcal{R}$ CQE instances in which $\mathcal{P}$ is an acyclic policy for $\mathcal{T}$, if a clash for a BUCQ q exists, then there exists a clash for q whose length depends only on the size of $\mathcal{P} \cup \mathcal{T} \cup \{q\}$.

Lemma 7. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance such that $\mathcal{P}$ is acyclic for $\mathcal{T}$, and let q be a BUCQ such that $q_i \in$ **BCQ**-$Cons(\mathcal{T} \cup \mathcal{A})$ for every $q_i \in q$. Then, $\mathcal{E} \models_{SC} q$ iff there exists no BCQ q' such that q' is a clash for q in $\mathcal{E}$ and Len(q') $\leq \ell$, where $\ell = m \cdot k^h$, m is the number of BCQs in q, k = MaxLenCQ($\mathcal{P}$), and h is the number of EDs in $\mathcal{P}$.*

We can now define an FO sentence and then prove that it provides a first-order rewriting of SC-entailment of BUCQs.

Definition 9. *Let $\mathcal{T}$ be a DL-Lite$_\mathcal{R}$ TBox, let $\mathcal{P}$ be an acyclic policy for $\mathcal{T}$, and let $q \in$ **BUCQ**. We define the FO sentence SC-Entailed($q, \mathcal{T}, \mathcal{P}$) as follows:*

$$\bigvee_{q^p \in \wp^-(q)} \Big(\bigwedge_{q_i \in q^p} \textit{PerfectRef}(q_i, \mathcal{T}) \wedge \bigwedge_{q_i \in q \setminus q^p} \neg \textit{PerfectRef}(q_i, \mathcal{T}) \wedge \bigwedge_{q'(\vec{x}) \in \mathcal{Q}} \neg \big(\exists \vec{x} \, (\textit{Clash}(q^p, q'(\vec{x}), \mathcal{T}, \mathcal{P}) \wedge \bigwedge_{x \in \vec{x}} Ind(x)) \big) \Big)$$

with $\wp^-(q) = \wp(q) \setminus \{\emptyset\}$, where $\wp(q)$ is the powerset of q, $\mathcal{Q}$ is the set of CQs defined over the predicates and constants occurring in $\{q\} \cup \mathcal{P} \cup \mathcal{T}$ and whose maximum length is $m \cdot k^h$, m is the number of BCQs in q, h is the number of EDs in $\mathcal{P}$, and k = MaxLenCQ($\mathcal{P}$).

Based on Lemma 6 and Lemma 7, we are able to prove the following crucial property for *SC-Entailed*($q, \mathcal{T}, \mathcal{P}$).

Lemma 8. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance such that $\mathcal{P}$ is an acyclic policy for $\mathcal{T}$, and let $q \in$ **BUCQ**. Then, SC-Entailed($q, \mathcal{T}, \mathcal{P}$) is a first-order rewriting of SC-entailment of q for $\mathcal{T}$ and $\mathcal{P}$.*

The above first-order rewritability property of SC-entailment of BUCQs immediately implies the next result.

Theorem 6. *SC-AREC[**BUCQ**] is in AC^0 in data complexity.*

Finally, given a BUCQ q, we define the sentence *IC-Entailed*($q, \mathcal{T}, \mathcal{P}$) as follows:

$$\bigvee_{q_i \in q} \textit{SC-Entailed}(q_i, \mathcal{T}, \mathcal{P})$$

It is then easy to prove the analogous of Lemma 8 (and Theorem 6) for *IC-Entailed*($q, \mathcal{T}, \mathcal{P}$) and IC-entailment.

Lemma 9. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance such that $\mathcal{P}$ is an acyclic policy for $\mathcal{T}$, and let $q \in$ **BUCQ**. Then, IC-Entailed($q, \mathcal{T}, \mathcal{P}$) is a first-order rewriting of IC-entailment of q for $\mathcal{T}$ and $\mathcal{P}$.*

Proof. The result follows immediately from Lemma 8 and from the fact that $\mathcal{E} \models_{\mathrm{IC}} q$ iff there exists a BCQ $q_i \in q$ such that $\mathcal{E} \models_{\mathrm{SC}} q_i$ (the latter property easily follows from Definition 5 and Theorem 1). □

Theorem 7. *IC-AREC[**BUCQ**] is in AC^0 in data complexity.*

6 Conclusions

The results given in this paper are summarized in Table 1. Beyond their theoretical connotation, our results for acyclic dependencies are particularly interesting for practical applications, since data complexity in these cases is the same as that for standard query answering over databases, and this paves the way for implementation through consolidated SQL technology. Moreover, the table shows that confidentiality is preserved in most of the cases that we have considered.

We posit that the epistemic nature of our framework makes it suited to being extended to incorporate user background knowledge, which can be modeled through appropriate epistemic formulas. The implications of this extension remain a subject for future investigation. Further possible development may explore ontology languages alternative to DL-Lite$_\mathcal{R}$, such as $\mathcal{EL}$ [Baader *et al.*, 2005] or the OWL 2 profiles [Motik *et al.*, 2012]. Additionally, extending the framework to accommodate preferences on data to be censored while ensuring compliance to the policy, as in [Cima *et al.*, 2021], and examining a dynamic context where censors filter responses based on previous answers, as explored in [Bonatti *et al.*, 2022], are paths for further research.

[1] W.l.o.g. we assume that the sets of existentially quantified variable symbols used by the CQs in $\mathcal{Q}$ are pairwise disjoint.

Acknowledgements

This work was partially supported by: projects FAIR (PE0000013) and SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the EU - NextGenerationEU; GLACIATION project funded by the EU (N. 101070141); ANTHEM (AdvaNced Technologies for Human-centrEd Medicine) project (CUP B53C22006700001) funded by the National Plan for NRRP Complementary Investments; the MUR PRIN 2022LA8XBH project Polar (POLicy specificAtion and enfoRcement for privacy-enhanced data management); and by the EU under the H2020-EU.2.1.1 project TAILOR (grant id. 952215).

References

[Baader *et al.*, 2005] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 364–369, 2005.

[Baader *et al.*, 2020] Franz Baader, Francesco Kriegel, Adrian Nuradiansyah, and Rafael Peñaloza. Computing compliant anonymisations of quantified ABoxes w.r.t. $\mathcal{EL}$ policies. In *Proc. of the 19th Int. Semantic Web Conf. (ISWC)*, volume 12506 of *Lecture Notes in Computer Science*, pages 3–20. Springer, 2020.

[Biskup and Bonatti, 2004] Joachim Biskup and Piero A. Bonatti. Controlled query evaluation for enforcing confidentiality in complete information systems. *Int. J. Inf. Sec.*, 3(1):14–27, 2004.

[Biskup and Weibert, 2008] Joachim Biskup and Torben Weibert. Keeping secrets in incomplete databases. *Int. J. Inf. Sec.*, 7(3):199–217, 2008.

[Biskup, 2000] Joachim Biskup. For unknown secrecies refusal is better than lying. *Data and Knowledge Engineering*, 33(1):1–23, 2000.

[Bonatti and Sauro, 2013] Piero A. Bonatti and Luigi Sauro. A confidentiality model for ontologies. In *Proc. of the 12th Int. Semantic Web Conf. (ISWC)*, pages 17–32, 2013.

[Bonatti *et al.*, 2022] Piero Bonatti, Gianluca Cima, Domenico Lembo, Lorenzo Marconi, Riccardo Rosati, Luigi Sauro, and Domenico Fabio Savo. Controlled query evaluation in OWL 2 QL: A "longest honeymoon" approach. In *Proc. of the 21st Int. Semantic Web Conf. (ISWC)*, volume 12922 of *Lecture Notes in Computer Science*, pages 428–444. Springer, 2022.

[Bonatti, 2022] Piero A. Bonatti. A false sense of security. *Artificial Intelligence*, 310, 2022.

[Calvanese *et al.*, 2007a] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. EQL-Lite: Effective first-order query processing in description logics. In *Proc. of the 20th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 274–279, 2007.

[Calvanese *et al.*, 2007b] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Automated Reasoning*, 39(3):385–429, 2007.

[Cima *et al.*, 2021] Gianluca Cima, Domenico Lembo, Lorenzo Marconi, Riccardo Rosati, and Domenico Fabio Savo. Controlled query evaluation over prioritized ontologies with expressive data protection policies. In *Proc. of the 20th Int. Semantic Web Conf. (ISWC)*, volume 12922 of *Lecture Notes in Computer Science*, pages 374–391. Springer, 2021.

[Cima *et al.*, 2024] Gianluca Cima, Domenico Lembo, Lorenzo Marconi, Riccardo Rosati, and Domenico Fabio Savo. Controlled query evaluation through epistemic dependencies. *CoRR*, abs/2405.02458, 2024.

[Console and Lenzerini, 2020] Marco Console and Maurizio Lenzerini. Epistemic integrity constraints for ontology-based data management. In *Proc. of the 37th AAAI Conf. on Artificial Intelligence (AAAI)*, pages 2790–2797. AAAI Press, 2020.

[Cuenca Grau *et al.*, 2013] Bernardo Cuenca Grau, Evgeny Kharlamov, Egor V. Kostylev, and Dmitriy Zheleznyakov. Controlled query evaluation over OWL 2 RL ontologies. In *Proc. of the 12th Int. Semantic Web Conf. (ISWC)*, pages 49–65, 2013.

[Cuenca Grau *et al.*, 2015] Bernardo Cuenca Grau, Evgeny Kharlamov, Egor V. Kostylev, and Dmitriy Zheleznyakov. Controlled query evaluation for datalog and OWL 2 profile ontologies. In *Proc. of the 24th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 2883–2889, 2015.

[European Union, 2016] European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016. *Official J. of the European Union*, L 119:48, 2016.

[Lembo *et al.*, 2019] Domenico Lembo, Riccardo Rosati, and Domenico Fabio Savo. Revisiting controlled query evaluation in description logics. In *Proc. of the 28th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 1786–1792, 2019.

[Motik *et al.*, 2012] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz. OWL 2 Web Ontology Language profiles (second edition). W3C Recommendation, World Wide Web Consortium, December 2012. Available at http://www.w3.org/TR/owl2-profiles/.

[Sicherman *et al.*, 1983] George L. Sicherman, Wiebren de Jonge, and Reind P. van de Riet. Answering queries without revealing secrets. *ACM Trans. on Database Systems*, 8(1):41–59, 1983.

[Vardi, 1982] Moshe Y. Vardi. The complexity of relational query languages. In *Proc. of the 14th ACM SIGACT Symp. on Theory of Computing (STOC)*, pages 137–146, 1982.